ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Scuola di Scienze
Dipartimento di Fisica e Astronomia
Corso di Laurea Magistrale in Fisica

MODELING TEMPORAL NETWORKS WITH DYNAMIC STOCHASTIC BLOCK MODELS

Relatore:
Prof. Fabrizio Lillo

Correlatore:
Dott. Piero Mazzarisi

Presentata da:
Federico Paltrinieri

Anno Accademico 2018/2019

Abstract

Osservando il recente interesse per le reti dinamiche temporali e l'ampio numero di campi di applicazione, questa tesi ha due principali propositi: primo, di analizzare alcuni modelli teorici di reti temporali, specialmente lo stochastic blockmodel dinamico, al fine di descrivere la dinamica di sistemi reali e fare previsioni. Il secondo proposito della tesi é quello di creare due nuovi modelli teorici, basati sulla teoria dei processi autoregressivi, dai quali inferire nuovi parametri dalle reti temporali, come la matrice di evoluzione di stato e una migliore stima della varianza del rumore del processo di evoluzione temporale. Infine, tutti i modelli sono testati su un data set interbancario: questi rivelano la presenza di un evento atteso che divide la rete temporale in due periodi distinti con differenti configurazioni e parametri.

Observing the recent interest for dynamic temporal networks and a large number of areas where they could be applied, this thesis has two major purposes: first to analyze existing theoretical temporal models, especially the dynamic stochastic block model, with the purpose to infer the underlying structure of the network and to predict its evolution. The second purpose is to propose two new theoretical models, based on the theory of auto-regressive processes, by which infer new parameters from temporal networks, such as the state evolution matrix and a better estimate of the variance of the process noise. Finally, all the models are applied to interbank data set: they reveal the presence of an expected event that divides the temporal networks into two different time-lapses with different configurations and parameters.

Contents

1 Introduction **3**
- 1.1 Graph theory and physics . . . 4
- 1.2 The topic of the thesis . . . 6
- 1.3 Review of the contents . . . 7

2 Time Series **10**
- 2.1 Autoregressive Model . . . 10
  - 2.1.1 Stationarity . . . 11
  - 2.1.2 Correlation . . . 11
  - 2.1.3 Autocorrelation function . . . 11
  - 2.1.4 AR(1) . . . 12
- 2.2 Vector Autoregressive Model . . . 13
  - 2.2.1 Properties: stationarity and moments . . . 14
- 2.3 Kalman Filter . . . 15
  - 2.3.1 Local trend model . . . 15
  - 2.3.2 The Kalman filter equations . . . 16
- 2.4 Extended Kalman Filter . . . 18

3 Stochastic Block Model **19**
- 3.1 Networks . . . 19
- 3.2 Model Definition . . . 22
- 3.3 Generative Model . . . 23
- 3.4 Inference Process . . . 27
  - 3.4.1 The spectral clustering method . . . 28
  - 3.4.2 The detectability problem . . . 29
- 3.5 Conclusion . . . 30

4 Dynamic Stochastic Block Model: Analysis and Numerical Simulations **32**
- 4.1 Xu and Hero Dynamic SBM . . . 33
  - 4.1.1 Generative model . . . 33
  - 4.1.2 Inference process . . . 34

4.1.3 Model analysis . . . 37
4.2 Numerical Simulations' Settings . . . 38
4.2.1 Analysis of the code . . . 38
4.2.2 Parameters setting . . . 39
4.2.3 Inference process and the metrics . . . 40
4.3 Analysis of the Numerical Results under Different Model Specifications . . . 41
4.3.1 Generative model: long-time simulations and absorbing states . . . 41
4.3.2 Groups fixed and a varying number of nodes . . . 44
4.3.3 Simulation study with varying number of groups . . . 53
4.3.4 Memory effect . . . 55
4.4 Conclusion . . . 57

5 Generalization of Xu and Hero Model to Auto-Regressive Processes 59
5.1 AR-SBM Model . . . 59
5.1.1 Model's hyperparameters . . . 60
5.2 VAR-SBM model . . . 62
5.2.1 Model's hyperparameters . . . 62
5.3 AR-SBM Model Results . . . 64
5.3.1 Network creation . . . 64
5.3.2 Inference of the model's hyperparameters . . . 67
5.4 VAR-SBM Model Results . . . 71
5.4.1 Generative model . . . 71
5.4.2 Inference of model's hyperparameters . . . 73
5.5 Comparison between AR-SBM and VAR-SBM Model . . . 77
5.5.1 MSE of $\boldsymbol{A}$, B, and Γ . . . 78
5.6 Conclusion . . . 81

6 Application to Interbank Network 82
6.1 Interbank Network and the e-MID . . . 82
6.1.1 The e-MID network . . . 83
6.2 Dataset . . . 84
6.2.1 Criteria applied to the dataset . . . 86
6.2.2 Inference Validation . . . 87
6.3 Result . . . 89
6.3.1 Xu and Hero model . . . 89
6.3.2 VAR-SBM Model . . . 95
6.3.3 Model comparison and conclusion . . . 102

7 Conclusion 104

A Computation of Kalman gain for non-diagonal matrix 106

Chapter 1

Introduction

Nowadays the amount of data we can produce and collect is growing faster than what we could only imagine a few years ago: World Wide Web is always expanding, social networks collect data from increasing numbers of users and from their interactions, databases of proteins or amino-acids are daily enlarged, numerical simulations run with millions of interacting particles. We give an idea of this enormous growth: the data traffic that will be produced in 2022 is estimated to be comparable with traffic produced between 1984 and 2016. We're talking about 4.8 ZettaBytes of data [36]. Focusing on Cern (European Organization for Nuclear Research), where is set the largest particles accelerator on Earth, all the experiments have generated more or less 49 PetaBytes of data in 2016 [8]. This number became 72 PetaBytes in 2017 [9]. This growth is related to the sensors' cost, which decreases in the past decade, implying a large diffusion of them in all the electronic devices we use.
Meanwhile, the processing power has grown to handle these data, even in daily devices: all the common laptops now work with multicore processors and frequently they have multicore threading GPU.

From a scientific point of view, this enormous amount of data has made it possible to study and to investigate new research areas. Several new analytical models describe the observed system's dynamic in these new fields and the same is for the related numerical algorithms. Graph theory is one of those that has grown most in last decades. This growth is due to the large progress in the computing power, which allows today to study very large systems. By this, many different study fields have started to employ graph theory's models and properties to analyze their research area: physics, biology, math, econometrics, computer science. Meanwhile, graph theory has been enriched by the analytical models used in all these fields, especially from the physical and mathematical models. In the following section, we will deepen this aspect.

1.1 Graph theory and physics

The interconnection between physics and graph theory is evident in many different fields of research of the physics: statistical physics, condensed matter physics, applied physics, to name a few. There are many systems of interest to scientists that are composed of individual parts or components linked together. The strength of graph theory lies in the possibility to describe these objects as nodes and links: examples include spin chains, vehicles traffic, aminoacids in proteins, the Internet, a collection of computers linked by data connections, and human societies.

The history of the graph theory is quite recent: in fact, the great majority of the notions and models have been formulated in the last 70 years. In the previous centuries, we can recall two main problems of the graph theory: the Euler's paper "The seven bridges of Konigsberg", written in 1736, describes the first mathematical problem solved with the graph theory. The problem could be stated as: "Does exist a path which crosses all the seven Konigsberg bridges only once?". Euler demonstrated that a similar path does not exist by creating a graph using the geographical chart of the city. He transformed every connected place in a node and every bridge in a link.

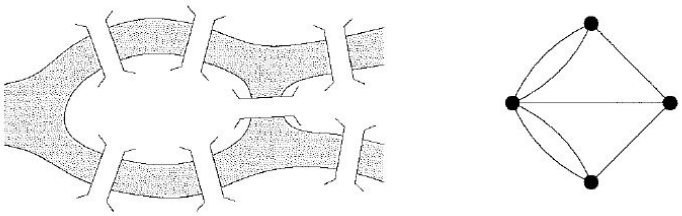

Figure 1.1: The Koningsberg bridges: on the left we observe the map of the city, on the right its graph representation

The second main problem is the famous *four color problem*, stated in 1852: is it possible to color any map using only four colors in such a way that adjacent regions are colored with different colors? The adjacent regions are those sharing a common boundary segment. The graph representation of this problem is the following one: each region is substituted with a node and each boundary between regions is replaced with a link between the respective nodes. Many demonstrations were proposed through the years, but the final and correct one arrived only in 1977. This demonstration is quite peculiar since the solution has not an analytical form: Kenneth Appel e Wolfgang Haken, the mathematicians who found the solution, used a computer algorithm to try all the possible combinations of colors on each possible topology of graph.
By these two problems, we observe that the solutions are obtained by transforming the studied systems into their network counterparts. This fact is well summarize by Newman: *a network is a simplified representation that reduces a system to an abstract structure capturing only the basics of connection patterns and little else* [31].

In the recent years, graph theory and physics become more and more intercorrelated: probably, statistical physics is the field more strictly connected with graph theory respect to the others. We review some examples by starting from the joining milestone, the Ising model. The Ising model, which is a benchmark for all the lattice models, is one of the statistical models used to describe the microscopical interaction in the matter. The Ising model considers the interactions between spins on a d-dimensional lattice. Its Hamiltonian is written as follows:

$$\mathcal{H} = -J \sum_{ij} \sigma_i \sigma_j - H \sum_{i=1}^{N} \sigma_i \tag{1.1}$$

where σ represents the spin and it assumes the values ± 1, J is the interaction term and H represents the external field. The interaction is considered only for nearest-neighbor spins. The analogy with the graph theory is self-evident: each spin in the Ising model could be replaced with a node, while the interactions between them are the network's links. In literature several graph analytical formulations exist to describe the Ising model [14] [16].
Continuing our analysis, one of the generalizations of the Ising model, the Potts model, is connected to the Max κ-cut problem, largely studied in the graph theory. Firstly we define the Potts model. This model is described by a lattice of spins and each of them can assume q different values or states. Thus, the Hamiltonian of the system could be written as follows:

$$\mathcal{H} = \sum_{ij} J_{ij}(1 - \delta(\sigma_i \sigma_j)) \tag{1.2}$$

where δ is the Kronecker delta function. Now we can focus on the Max κ-cut problem. This problem could be stated as follows: for any graph $G = (V; E)$, the problem tries to split the nodes V into κ parts such that the number of links between different parts is maximized [27]. The Max κ-cut problem could be applied to each family of random graphs. In the case of sparse, inhomogeneous stochastic networks and in the large degree limit, the asymptotic behavior of the leading term of the Max κ-cut is a variational problem involving the ground state of a constrained inhomogeneous Potts spin glass [27] [18] [17].
The connection between the Potts model and graph theory could be extended to problems of the optimal inference of the group assignment. In the following example, we review a specific application of the stochastic block model, which falls in the class of the random networks originated by the Erdos-Renyi Model, whose particularity is to relation the probabilities of linking between nodes and the structure of the communities in the network through the affinity matrix, which entries are the probabilities of linking between different groups. This model is the benchmark for thesis and ti will be further detailed in the followings. The analysis of the stochastic block model (SBM) made by Decelle et al. [13] considers the probability $P(G, \{q_i\}|\boldsymbol{\theta})$ that the SBM generates a graph G, with N nodes, adjacency matrix W with group assignment $\{q_i\}$, conditioned on the parameters of the model θ. Employing the Bayes' rule,

Decelle et al. explicit the probability distribution over the group assignments by knowing the graph G and the parameters θ:

$$P(\{q_i\}|G,\boldsymbol{\theta}) = \frac{P(G,\{q_i\}|\boldsymbol{\theta})}{\sum_{t_i} P(G,\{t_i\}|\boldsymbol{\theta})} \tag{1.3}$$

They notice that this distribution is the Boltzmann distribution of the generalized Potts model with Hamiltonian:

$$\mathcal{H}(\{q_i\}|G,\boldsymbol{\theta}) = -\sum_i \log n_{q_i} - \sum_{i\neq j}\left[W_{ij}\log c_{q_i,q_j} + (1-W_{ij})\log\left(1-\frac{c_{q_i,q_j}}{N}\right)\right] \tag{1.4}$$

where n_{q_i} is the number of nodes in the group q_i and the c_{q_i,q_j} are the entries of rescaled affinity matrix. The analogy is between the labels q_i and the spins and between the number of nodes in the groups n_{q_i} and the local magnetic fields [13].

By a few examples, we explicited the close connection between physics and graph theory. In the following sections, we analyze the aim of this thesis and we summarize its contents.

1.2 The topic of the thesis

In the previous section, we have analyzed different applications of graph theory, especially by connecting them to physics. In this section, we want to explicit the aim of our study developed through the thesis. We can start this operation by considering the thesis title: "modeling temporal networks with dynamic stochastic block models". By reading this title, we immediately have to answer a question: what is it a temporal network? Temporal networks are all those systems that evolve and are composed of entities that have some kinds of relations between themselves. The most important fact is that the relations between the entities evolve over time. Anyway, this definition remains quite general, thus we also explain what is a temporal network using some examples. Probably, the easiest one concerns social relations. We consider to imagine all the people we know: some of them are strongly bound to us (like our best friends, or our parents), others are people we see every day and we know them because we share something (like our classmates, or our colleagues), others are simply acquaintances. We imagine tying all the couples of people that have some kind of relationship among these people we know. The result we obtain is the network of the social relations among the people of our life, where the people are the nodes and the relations are links that join them. If we would have repeated the same process a few years ago, probably some of them were not still in our life, others should still have known each other. Hence, the network we would have built a few years ago is not the network we build today, which is not the network we will build in a few years. If we put together all these networks, we obtain the temporal network which describes all the relations between all the people we have known in all our life. This process is

similar to the chronological collection of all the snapshots of the relations we acquired through each period of our life. But examples of temporal networks exist in any field: if we consider the pattern of the hyperlinking in the WWW, it evolves constantly every second: some sites are deleted, others are created, others are joined with a hyperlink. Thus, we have obtained a solid idea of what a temporal network is, but this definition remains purely qualitative. When modeling a temporal network, we have to make important choices to represent them. Through the previous example, we explicit this problem: are the relations between the people all equals or some of them are more important than the others? If some people share more relations between themselves respect to the others, should we consider them as a group? How many groups are in the networks? With these questions, we have pointed out some problems when modeling temporal networks. Not all the relations have the same weight (a weighted network is a network where different values are associated with the edges to give different relevances to the studied relations, i.e. a weight for each of relation) : this fact can be considered as not when modeling a temporal network. Furthermore, what is a group in a network? Do the groups evolve and how they evolve? How the dynamic of the relations between different groups evolve through time? These are non-trivial problems to solve since we aim to solve them by knowing only the patterns of the relations between the entities in the temporal network and nothing more: this information is called the temporal adjacency matrix of the temporal network. How to do you this? Employing the second parts of the title of our thesis, the dynamic stochastic block models. This class of network models has recently become a benchmark for the analysis of temporal networks, thanks to its easy parametrizations. Furthermore, the parametrization of stochastic block models in general (both static and dynamic) give us important information about the network, like which is the relations between entities of the same groups and which is the relations between different groups. One of the key properties of the stochastic block models is that it does not assume a pre-established structure between the groups of the network: this allows the model to infer the information of the network not for what we expect it would be like other models do. Concluding, we aim to study the modelization of temporal networks and to understand how the different groups interact between themselves through the time in the network using the dynamic stochastic block model.

1.3 Review of the contents

The thesis is organized in 7 chapters, the first of which is this one. It possible to divide the thesis into two main parts by grouping the chapters: Chapter 2, Chapter 3, and part of the fourth one form the theoretical background of the thesis. In these chapters, we discuss mainly two subjects: time series analysis and the stochastic block model, both the static and dynamic models. We review a particular dynamic stochastic block model proposed by Xu and Hero [40] and we analyze their model using numerical simulations. The other chapters describe our original work and results: we propose two new dynamic stochastic block models based on autoregressive models discussed in Chapter 2. Hence, we show the results of numerical

simulations related to the newly proposed models. Finally, we test Xu and Hero model and our new models on a real network, i.e. the interbank market. Deepening the analysis chapter by chapter:

- In Chapter 2, we discuss some elements of time series theory. The chapter is divided into two main approaches in the time series field: the autoregressive model theory and the filters' theory. The analytical models we review are the autoregressive model (AR), the vector autoregressive model (VAR), the Kalman filter, and the extended Kalman filter (EKF). The AR model and the VAR model will be employed in Chapter 5, where we propose two new dynamic stochastic block models based on them. The EKF will be employed in Chapter 4 in the Xu and Hero model. This chapter is based in part on the book *Analysis of Financial Time Series* [38], in part on *Time Series Analysis by State Space Methods* [15].

- In Chapter 3 we discuss the stochastic block model (SBM) in its simplest version [11] [10]. Since the chapter is based on graph theory, we expose briefly what is a network and the main concepts of graph theory [31]. Thus, we explore both the generative model and the inference process of the stochastic block model. We conclude by reviewing the detectability problem following the paper of Decelle et al. [13].

- In Chapter 4 we examine the dynamic stochastic block models proposed by Xu and Hero [40] and we focus on the algorithm proposed for the inference process on dynamic networks. Thus, we show the results of the numerical simulations obtained using this algorithm: we explore the limit of this model by setting the model's parameters out of range used by the model authors. By this, we point out some drawbacks of the Xu and Hero model. All the numerical results obtained in this chapter are original.

- In Chapter 5 we propose two new dynamic stochastic block models, the AR-SBM model and the VAR-SBM model, based on the AR process and on the VAR process. These models are thought to solve the problems found in Xu and Hero model in Chapter 4. Hence, we focus on the hyperparameters required by these models in the inference process on dynamic networks. We conclude by showing the results of the numerical simulations for the inference process obtained from both the models. The simulations are set to compare the performances of the new models with the Xu and Hero model. All the results obtained in this chapter are original.

- In Chapter 6 we test our models on a real network, the interbank market. We discuss the features of the interbank market [26] [12] [7] [5] [20] and we focus on the e-MID [5], which is one the interbank market, and the one we employ to generate the dynamic network. The analysis of the e-MID network is based on the paper of Barucca and Lillo [5], where they study the community structure of the banks in the e-MID market in the period 2010-2014. They focus their attention on the effect of the LTROs on the

e-MID, analyzing if these measures used by the European Central Bank affected the behaviors in the market of the banks. Thus, we show our results obtained from the numerical simulations of the Xu and Hero model and of the VAR-SBM model on the e-MID network. We observe a bipartite structure of the network in the whole studied period. We also notice a change in the behavior of the banks due to the LTROs. At the end, we compare the results obtained with the different models and we discuss the literature results [5] [20].

Chapter 2

Time Series

This chapter describes one of the two theoretical backgrounds of our work. The analysis of time series includes a lot of models and is widely employed in physics as much as in financial econometrics. Time series analysis is a fundamental tool when the evolution of the state of the system under investigation can be described by a time series. When we assume linear dependence between the current state and the past states of the system, we can describe the system dynamics by means of an autoregressive process. In this chapter, we review the autoregressive model of order one for both univariate and multivariate time series, in particular we study the main properties and the conditions for stationarity. Sometimes, we do not have access to the state of the system with infinite precision, but any measurement is associated with some observation noise. In this case, Kalman filter helps in estimating the state of the system as well as its linear dynamics. Moreover, this estimation method can be further extended to the case of non-linear dependences, a method known as extended Kalman filter. We review both statistical filtering methods, which represent one building block of our research presented in the next chapters. The following analysis is based in part on the book *Analysis of Financial Time Series* [38], in part on *Time Series Analysis by State Space Methods* [15].

2.1 Autoregressive Model

Time-series modeling, analysis, and forecasting have fundamental importance in many research fields. Any time the state of a system can be described by a given set of data points, its dynamical evolution is represented by a time series. Hence, models and analysis of time series are of fundamental importance to extract information from observed data. Autoregressive models are probably the most popular and adopted models of time series. An autoregressive model is a discrete-time random process where the output variable is a linear combination of the previous observations (up to a given time lag) and depends also on a stochastic term. Thus, an autoregressive model is to all effects a stochastic difference equation. A generic autoregressive process can be characterized by its stationarity condition and both the autocorrelation and

cross-correlation function (the last one in the case of a vector autoregressive process where more than one output variable is present).

2.1.1 Stationarity

A time-series $\{r_t\}$ is said to be *strictly stationary* if the joint distribution of $(r_{t_1}, \ldots, r_{t_k})$ is identical to that of $(r_{t_1+t}, \ldots, r_{t_k+t})$ for all t, where k is an arbitrary positive integer and $(t_1, \ldots, t_k)$ is a collection of k positive integers. Hence, this is equivalent to say that the joint distribution of $(r_{t_1}, \ldots, r_{t_k})$ is invariant under time shift. This a very strong condition that is hard to verify empirically. A weaker version of stationarity can be considered and it is called *weakly stationarity*. A time-series $\{r_t\}$ is *weakly stationary* if both the mean of r_t and the covariance between r_t and r_{t-l} are time-invariant, where l is an arbitrary integer. Thus, a time series $\{r_t\}$ is weakly stationary if $E(r_t) = \mu$ and $Cov(r_t, r_{t-l}) = \gamma_l \forall t$, given $l \in \mathbb{Z}$.
In the condition of weak stationarity, we assume implicitly that the first two moments of r_t are finite.

2.1.2 Correlation

The correlation coefficient between two random variables X and Y is defined as:

$$\rho_{x,y} = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}} = \frac{E[(X-\mu_x)(Y-\mu_y)]}{\sqrt{E(X-\mu_x)^2 E(Y-\mu_y)^2}} \tag{2.1}$$

where μ_x and μ_y are the mean of X and Y, respectively, and it is assumed that the variances exist. This coefficient measures the linear dependence between X and Y. It is $-1 \leq \rho \leq 1$ and $\rho_{x,y} = \rho_{y,x}$. If two random variables are uncorrelated $\rho_{x,y} = 0$.

When two random variables X and Y are described by two time series $\{x_t\}_{t=1,\ldots,T}$ and $\{y_t\}_{t=1,\ldots,T}$, the correlation can be estimated by:

$$\hat{\rho}_{x,y} = \frac{\sum_{t=1}^{T}(x_t - \bar{x})(y_t - \bar{y})}{\sqrt{\sum_{t=1}^{T}(x_t - \bar{x})^2 \sum_{t=1}^{T}(y_t - \bar{y})^2}} \tag{2.2}$$

where $\bar{x} = \frac{1}{T}\sum_{t=1}^{T} x_i$ and $\bar{y} = \frac{1}{T}\sum_{t=1}^{T} y_i$ are the sample mean of the two time series, respectively.

2.1.3 Autocorrelation function

To analyze the linear dependence between r_t and its past values $r_{t-\ell}$, the concept of correlation is generalized to autocorrelation. The correlation coefficient between r_t and $r_{t-\ell}$ is called $lag-\ell$

autocorrelation of r_t and is denoted by ρ_ℓ. Specifically, we define:

$$\rho_\ell = \frac{Cov(r_t, r_{t-\ell})}{\sqrt{Var(r_t)Var(r_{t-\ell})}} = \frac{Cov(r_t, r_{t-\ell})}{Var(r_t)} = \frac{\gamma_\ell}{\gamma_0} \tag{2.3}$$

where $Var(r_t) = Var(r_{t-\ell})$ for the assumption of weakly stationarity. From the definition, we have $\rho_0 = 1$, $\rho_\ell = \rho_{-\ell}$, and $-1 \leq \rho_\ell \leq 1$. For a given sample of $\{r_t\}_{t=1}^T$, the $lag - \ell$ sample autocorrelation of r_t is:

$$\hat{\rho}_\ell = \frac{\sum_{t=\ell+1}^T (r_t - \bar{r})(r_{t-\ell} - \bar{r})}{\sum_{t=1}^T (r_t - \bar{r})^2} \tag{2.4}$$

2.1.4 AR(1)

The autoregressive process of order one is called $AR(1)$. The AR(1) process describes a random variable having Markov property and linear dependence from its previous value, and an innovation term given by a white noise, thus resulting in a sequence of independent and identically distributed random variables with zero mean and finite variance. Once defined the white noise, we can describe the $AR(1)$ process:

$$r_t = \phi_0 + \phi_1 r_{t-1} + a_t \tag{2.5}$$

where $\{a_t\}$ is assumed to be a white noise series with mean zero and variance σ_a^2. a_t is usually called *innovation* or *shock* of the time series. The AR(1) model ha several properties, like:

$$E[(r_t|r_{t-1})] = \phi_0 + \phi_1 r_{t-1} \quad Var[(r_t|r_{t-1})] = Var[(a_t)] = \sigma_a^2 \tag{2.6}$$

That is, given the past term of the series r_{t-1}, the current term is centered around $\phi_0 + \phi_1 r_{t-1}$ with variance σ_a^2. This is the Markov property meaning that conditional on r_{t-1}, the term r_t is independent of $r_{t-\ell}$ for $\ell > 1$.

Properties of AR(1) Model

Assuming that the series is weakly stationary, we have $E(r_t) = \mu, Var(r_t) = \gamma_0$ and $Cov(r_t, r_{t-l}) = \gamma_l$. We can easily obtain the mean, variance, and autocorrelation of the process as follows. Taking the expectation of Eq. 2.5, and because $E(a_t) = 0$, we obtain:

$$E(r_t) = \phi_0 + \phi_1 E(r_{t-1}) \tag{2.7}$$

Under the stationarity condition, $E(r_t) = E(r_{t-1}) = \mu$, so:

$$\mu = \phi_0 + \phi_1 \mu \implies \mu = \frac{\phi_0}{1 - \phi_1} \tag{2.8}$$

This result has two implications for r_t. First, the mean of r_t exists if $\phi_1 \neq 1$. Second, the mean of r_t is zero if and only if $\phi_0 = 0$. Thus, for a stationary AR(1) process, the constant term ϕ_0 is related to the mean of r_t via $\phi_0 = (1 - \phi_1)\mu$.
To compute the variance of the process, we can subtract the mean out. The model centered around the zero is:

$$r_t - \mu = \phi_1(r_{t-1} - \mu) + a_t \tag{2.9}$$

Because of the independence of the time series $\{a_t\}$, it is $E[(r_{t-1} - \mu)a_t] = 0$. Hence, by taking the square of Eq. 2.9, and considering the expectation of the square, it is trivially:

$$Var(r_t) = \phi_1^2 Var(r_{t-1}) + \sigma_a^2 \tag{2.10}$$

where σ_a^2 is the variance of a_t. Under the stationarity assumption, $Var(r_t) = Var(r_{t-1})$. It follows:

$$Var(r_t) = \frac{\sigma_a^2}{1 - \phi_1^2} \tag{2.11}$$

provided that $\phi_1^2 < 1$. The requirement of $\phi_1^2 < 1$ results from the fact that the variance of a random variable is bounded and non-negative [38].

In conclusion, the AR(1) model describes a state variable with Markov properties of order one (i.e. it depends linearly from its previous state at time lag 1) and having a stationary regime when some conditions (described above) for parameters are fulfilled. In the following, we describe the evolution of link probability between and inside communities of a dynamic network using an autoregressive process AR(1) to study its persistence pattern. Hence, as will be clear below, Eq. 2.5 and 2.11 will be used in the inference of model parameters.

2.2 Vector Autoregressive Model

The Vector Autoregressive Model (VAR), which is an extension of the Autoregressive Model 2.1, is probably one of the most known model for multivariate time series. Similarly to the AR(1) process, the VAR(1) model is a discrete-time random process, but for multivariate time series.

We can define a multivariate time series of dimension $k > 1$, named as $\boldsymbol{r}_t$, a VAR(1) process if the equation that describes the process is:

$$\boldsymbol{r}_t = \boldsymbol{\phi}_0 + \boldsymbol{\Phi}\boldsymbol{r}_{t-1} + \boldsymbol{a}_t \tag{2.12}$$

Where:

- $\boldsymbol{\phi}_0$ is a k-dimensional vector;

- $\mathbf{\Phi}$ is a $k \times k$ matrix;
- $\{\boldsymbol{a}_t\}$ is a vector of serially uncorrelated random vectors, with mean equal to zero and covariance matrix $\mathbf{\Sigma}$;

To explicit an example of the model, we can consider a bivariate case, where $k = 2$. In this case, the Eq. 2.12 becomes:

$$\begin{cases} r_{1t} = \phi_{10} + \Phi_{11} r_{1,t-1} + \Phi_{12} r_{2,t-1} + a_{1t} \\ r_{2t} = \phi_{20} + \Phi_{21} r_{1,t-1} + \Phi_{22} r_{2,t-1} + a_{2t} \end{cases} \tag{2.13}$$

This couple of equations shows that each series has a conditional effect on the other. This effect depends on the anti diagonal entries of the matrix $\mathbf{\Phi}$, named as Φ_{12} and Φ_{21}. If we set both to zero, the two series r_{1t} and r_{2t} are not coupled anymore.

2.2.1 Properties: stationarity and moments

Let us assume the model described by Eq. 2.12 is weakly stationary. We can study the properties of the VAR(1) Model with similar considerations we used for AR(1) model. Since the expected value of a_t is zero, it follows:

$$E(\boldsymbol{r}_t) = \boldsymbol{\phi}_0 + \mathbf{\Phi} E(\boldsymbol{r}_{t-1}) \tag{2.14}$$

By assuming stationarity, $E(\boldsymbol{r_t})$ results as time-invariant, thus:

$$\boldsymbol{\mu} = (\boldsymbol{I} - \mathbf{\Phi})^{-1} \boldsymbol{\phi}_0 \tag{2.15}$$

Where $\boldsymbol{I}$ is the k-dimensional identity matrix.
By applying a detrending procedure to the time series in Eq. 2.12, we obtain:

$$\tilde{\boldsymbol{r}}_t = \mathbf{\Phi} \tilde{\boldsymbol{r}}_{t-1} + \boldsymbol{a}_t \tag{2.16}$$

By iterating Eq. 2.16, the same equations reads as:

$$\tilde{\boldsymbol{r}}_t = \boldsymbol{a}_t + \mathbf{\Phi} \boldsymbol{a}_{t-1} + \mathbf{\Phi}^2 \boldsymbol{a}_{t-2} + \mathbf{\Phi}^3 \boldsymbol{a}_{t-3} + \dots \tag{2.17}$$

Analyzing this equation, we can observe that, in VAR(1) Model, $Cov(\mathbf{a}_t, \mathbf{a}_{t-1}) = 0$, since $\boldsymbol{a}_t$ is serially uncorrelated. Furthermore, we can notice:

$$Cov(\boldsymbol{r}_t, \boldsymbol{a}_t) = \mathbf{\Sigma} \tag{2.18}$$

where $\mathbf{\Sigma}$ is the covariance matrix. This result can be obtained by multiplying Eq. 2.17 by the transpose of $\mathbf{a}_t$ and taking its expectation.

We can also calculate the lag-l cross-covariance of $\boldsymbol{r}_t$:

$$\begin{aligned}
\tilde{\boldsymbol{r}}_t &= \boldsymbol{\Phi}\tilde{\boldsymbol{r}}_{t-1} + \boldsymbol{a}_t \\
\tilde{\boldsymbol{r}}_t\tilde{\boldsymbol{r}}'_{t-l} &= \boldsymbol{\Phi}\tilde{\boldsymbol{r}}_{t-1}\tilde{\boldsymbol{r}}'_{t-l} + \boldsymbol{a}_t\tilde{\boldsymbol{r}}'_{t-l} \\
E(\tilde{\boldsymbol{r}}_t\tilde{\boldsymbol{r}}'_{t-l}) &= E(\boldsymbol{\Phi}\tilde{\boldsymbol{r}}_{t-1}\tilde{\boldsymbol{r}}'_{t-l}) + E(\boldsymbol{a}_t\tilde{\boldsymbol{r}}'_{t-l}) \\
\boldsymbol{\Gamma}_l &= \boldsymbol{\Phi}\boldsymbol{\Gamma}_{l-1}
\end{aligned}$$

Where $\boldsymbol{\Gamma}_l = E(\tilde{\boldsymbol{r}}_t\tilde{\boldsymbol{r}}'_{t-l})$ is the lag-l cross-covariance. We use the result $E(\boldsymbol{a}_t\tilde{\boldsymbol{r}}'_{t-l}) = Cov(\boldsymbol{a}_t, \boldsymbol{r}_{t-l} = 0) =$ for $l > 0$.

2.3 Kalman Filter

In many systems, the state variables are not known exactly but their measurements contain some statistical noise. In this case, Kalman filtering is an algorithm that uses a more than one measurement observed over time and it can be applied to produce estimates of the unknown state variables that tend to be more accurate than those based on a single measurement alone. In the following, we describe how it works.

2.3.1 Local trend model

Let us consider the univariate time series y_t satisfying:

$$y_t = \mu_t + z_t \quad z_t \sim N(0, \sigma_z^2) \tag{2.19}$$

$$\mu_{t+1} = \mu_t + v_t \quad v_t \sim N(0, \sigma_v^2) \tag{2.20}$$

where $\{z_t\}$ and $\{v_t\}$ are two independent Gaussian white noise series and $t = 1, \ldots, T$. Specifically, z_t represents the statistical noise associated with measurements whereas v_t represents the innovation of the process. The initial value μ_1 is either given or follows a known distribution, and it is independent of $\{z_t\}$ and $\{v_t\}$ for $t > 0$. Here μ_t is a pure random walk with initial value μ_1 and y_t is an observed version of μ_t with added noise z_t. Here μ_t is not directly observable, and y_t is the observed data with observation noise v_t.

The model in Eq. 2.19 and 2.20 is a special linear Gaussian state-space model. The variable μ_t is called the *state* of the system at time t and is not directly observed. Eq. 2.19 provides the link between the data y_t and the state μ_t and is called the *observation equation* with *measurement noise* z_t. Eq. 2.20 governs the time evolution of the state variable and is the *state equation* or *state transition equation* with innovation v_t.

The Kalman filter procedure aims to infer properties of the state μ_t for the data $\{y_t|t = 1, \ldots, T\}$ and the model. The type of inference we are interested in is *filtering*, i.e. how to estimate the state variable μ_t given the measurements up to time t $F_t = \{y_1, \ldots, y_t\}$ or, equivalently, how to remove the measurement errors from the observations.

2.3.2 The Kalman filter equations

The goal of the *Kalman filter* is to update the knowledge of the state variable recursively when a new data point becomes available. That is, knowing the conditional distribution of μ_t, F_{t-1} and the new data y_t, we would like to obtain the conditional distribution of μ_t given F_t.

To describe the inference more precisely, we introduce the following notation. Let $\mu_{t|j} = E(\mu_t|F_j)$ and $\Sigma_{t|j} = Var(\mu_t|F_j)$ be, respectively, the conditional mean and variance of μ_t given F_j. Similarly, $y_{t|j}$ denotes the conditional mean of y_t given F_j. Furthermore, let $v_t = y_t - y_{t|t-1}$ and $V_t = Var(v_t|F_{t-1})$ be the 1-step-ahead forecast error and its variance of y_t given F_{t-1}. Note that the forecast error v_t is independent of F_{t-1} so that the conditional variance is the same as the unconditional variance; that is, $Var(v_t|F_{t-1}) = Var(v_t)$. From Eq. 2.19,

$$y_{t|t-1} = E(y_t|F_{t-1}) = E(\mu_t + e_t|F_{t-1}) = E(\mu_t|F_{t-1}) = \mu_{t|t-1} \tag{2.21}$$

Consequently:

$$v_t = y_t - y_{t|t-1} = y_t - \mu_{t|t-1} \tag{2.22}$$

and

$$\begin{aligned} V_t &= Var(y_t - \mu_{t|t-1}|F_{t-1}) \\ &= Var(\mu_t + e_t - \mu_{t|t-1}|F_{t-1}) \\ &= Var(\mu_t - \mu_{t|t-1}|F_{t-1}) + Var(e_t|F_{t-1}) \\ &= \Sigma_{t|t-1} + \sigma_e^2 \end{aligned} \tag{2.23}$$

It is also easy to see that

$$E(v_t) = E[E(v_t|F_{t-1})] = E[E(y_t - y_{t|t-1}|F_{t-1})] = E[y_{t|t-1} - y_{t|t-1}] = 0 \tag{2.24}$$

$$Cov(v_t, y_j) = E(v_t y_j) = E[E(v_t y_j|F_{t-1})] = E[y_j E(v_t|F_{t-1})] = 0 \quad j < t \tag{2.25}$$

Thus, as expected, the 1-step-ahead forecast error is uncorrelated with y_t for $j < t$.

The conditional distribution of v_t given F_{t-1} is normal with mean zero and variance given by Eq. 2.23, and that of μ_t given F_{t-1} is also normal with mean $\mu_{t|t-1}$ and variance $\Sigma_{t|t-1}$. Furthermore, the joint distribution of $(\mu_t, v_t)'$ given F_{t-1} is also normal. Thus, what remains to be solved is the conditional covariance between μ_t and v_t given F_{t-1}. From the definition,

$$\begin{aligned} Cov(\mu_t, v_t|F_{t-1}) &= E(\mu_t v_t|F_{t-1}) = E[\mu_t(y_t - \mu_{t|t-1})|F_{t-1}] \\ &= E[\mu_t(\mu_t + e_t - \mu_{t|t-1})|F_{t-1}] \\ &= E[\mu_t(\mu_t - \mu_{t|t-1}|F_{t-1}] + E[\mu_t e_t|F_{t-1}] \\ &= E[(\mu_t - \mu_{t|t-1})^2|F_{t-1}] = Var(\mu_t|F_{t-1}) = \Sigma_{t|t-1} \end{aligned}$$

Putting the results together, we have:

$$\begin{bmatrix} \mu_t \\ v_t \end{bmatrix}_{F_{t-1}} \sim N\left(\begin{bmatrix} \mu_{t|t-1} \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_{t|t-1} & \Sigma_{t|t-1} \\ \Sigma_{t|t-1} & V_t \end{bmatrix} \right) \tag{2.26}$$

The conditional distribution of μ_t given F_t is normal with mean and variance:

$$\mu_{t|t} = \mu_{t|t-1} + \frac{\Sigma_{t|t-1} v_t}{V_t} = \mu_{t|t-1} + K_t v_t \tag{2.27}$$

$$\Sigma_{t|t} = \Sigma_{t|t-1} - \frac{\Sigma^2_{t|t-1}}{V_t} = \Sigma_{t|t-1}(1 - K_t) \tag{2.28}$$

where K_t is referred to as the *Kalman gain*, which is the regression coefficient of $\mu_{t|t}$ on v_t.

We can make use of the knowledge of μ_t given F_t to predict μ_{t+1}:

$$\mu_{t+1|t} = E(\mu_t + \eta_t | F_t) = E(\mu_t | F_t) = \mu_{t|t} \tag{2.29}$$

$$\Sigma_{t+1|t} = Var(\mu_{t+1} | F_t) = Var(\mu_t | F_t) + Var(\eta_t) = \Sigma_{t|t} + \sigma^2_\eta \tag{2.30}$$

Once the new data y_{t+1} is observed, one can repeat the above procedure to update knowledge of μ_{t+1}. This is the *Kalman filter* algorithm proposed by Kalman in 1960. [28]

In summary, putting the equations together and conditioning on the initial assumption that μ_1 is distributed as $N(\mu_{1|0}, \Sigma_{1|0})$, the Kalman filter for the local trend model is as follows [38]:

$$v_t = y_t - \mu_{t|t-1} \tag{2.31}$$

$$V_t = \Sigma_{t|t-1} + \sigma^2_e \tag{2.32}$$

$$K_t = \frac{\Sigma_{t|t-1}}{V_t} \tag{2.33}$$

$$\mu_{t+1|t} = \mu_{t|t-1} + K_t v_t \tag{2.34}$$

$$\Sigma_{t+1|t} = \Sigma_{t|t-1}(1 - K_t) + \sigma^2_\eta \tag{2.35}$$

The Kalman filter is useful for linear problems, but when the underlying dynamics for the state variable is non linear, or the observation itself depends non linearly from the state variable, we need to generalize the filtering procedure. Thus we need a non-linear filter to solve this problem: one solution is the Extended Kalman Filter.

2.4 Extended Kalman Filter

The extended Kalman filter is based on the idea of linearising the observation and state equations and then applying the Kalman filter to the resulting linearised model. The model equations are:

$$y_t = h(\mu_t) + z_t \tag{2.36}$$

$$\mu_t = F_t(\mu_{t-1}) + v_t \tag{2.37}$$

for $t = 1, \ldots, T$, where $F_t(\mu_t)$ and $h(\mu_t)$ are differentiable functions of μ_t and where the random noises z_t and v_t are serially and mutually uncorrelated with mean zero and variance σ_z and σ_v, respectively.
The initial state is a random vector with mean μ_1 and covariance matrix Γ_1 and is uncorrelated with all the disturbances.
Due to the non-linearity of $h(\mu_t)$, we apply the extended Kalman filter (EKF), which linearises the dynamics of the predicted state and provides a *near-optimal* estimate of μ_t.
The EKF equations are specified by Eq. 2.36 and 2.37. The predicted state estimate is:

$$\mu_{t|t-1} = F_t(\mu_{t-1|t-1}) \tag{2.38}$$

and the predicted covariance estimate R_t is:

$$R_{t|t-1} = F_t R_{t-1|t-1}(F_t)^T + \sigma_v \tag{2.39}$$

Thus, we define:

$$H_t = \left.\frac{\partial h(\mu_t)}{\partial \mu_t}\right|_{\mu_t = \mu_{t|t-1}} \tag{2.40}$$

We note that H_t is evaluated at time $t-1$, since $\mu_{t|t-1}$ is a function of $y_1, \ldots, y_{t-1}$. Hence, using the linearization of the function $h(\mu_t)$ with the first term of the Taylor expansion, we can write the near-optimal Kalman gain:

$$K_t = R_{t|t-1}(H_t)^T[H_t R_{t|t-1}(H_t)^T + \sigma_z]^{-1} \tag{2.41}$$

from which the updated state estimate:

$$\mu_{t|t} = \mu_{t|t-1} + K_t[y_t - h(\mu_{t|t-1})] \tag{2.42}$$

and the updated covariance estimate:

$$R_{t|t} = (I - K_t H_t)R_{t|t-1} \tag{2.43}$$

In the following, we study the problem of inferring both the community structure and the link probability among groups of dynamic networks whose evolution is described by a dynamic generalization of the stochastic block model [40]. The inference of this model is characterized by two aspects: first, the presence of measurement noise for the link probabilities among groups, and second, the nonlinear mapping between link probability and the corresponding state variable, that is a real variable whereas the probability is defined within the unit interval. Hence, EKF is used for inferring the stochastic block model and, as a consequence, it represents one of the building blocks of the estimation method.

Chapter 3

Stochastic Block Model

The stochastic blockmodel (SBM) is one of the most used models if we want to generate blocks, communities, or groups in networks. This model falls in the class of the random networks, originated by the Erdos-Renyi Model. It was first studied in mathematical sociology by Holland, Laskey, and Leinhardt and by Wang and Wong in 1983 – 1987 [22] [39]. This model is widely employed as a generative model: in fact, it allows us to generate a very large number of different block structures. This versatility, combined with analytical tractability, has made the blockmodel a popular tool in several contexts [29].
Many generalizations were proposed through the years, like the labeled SBM [21], the Degree-corrected SBM [29], or the overlapping SBM [2].

Our purpose is to study the original SBM: we analyze the versatility of the model as a generative model and we briefly review the methods used for the inference process. Among these methods, we focus on the spectral clustering. Finally, following the analysis of Decelle et al. [13], we discuss the problem of the detectability, which is the condition to extract informations on the network communities by studying only the structure of links of the graph. This chapter is divided into four parts: in the first part we review the main concepts of the graph theory, the second one is model definition (bases on the lessons of prof. Clauset [10] [11]), the third one is the analysis of the generative model [10] and the fourth one is the analysis of the inference process [11] [13].

3.1 Networks

The aim of this section is to introduce the main concepts of network theory we will employ through the thesis.

The first element to introduce is the definition of graph, also referred as network: *a graph is a collection of vertices joined by edges*[31]. The vertices and edges are also called nodes or

links (these are the terms will use in the following chapters). The connection between nodes can be realized with a single link or with many links: the latter case is called *multiedges* connection. In some cases, a nodes is connected to itself with a link: this link is called *self-loop*. In the following we will analyze only *simple network*, i.e. graph without self-loops or multiedges. Another important notion in graph theory is the concept of *communities*, also called clusters, modules or groups. Communities «groups of vertices which probably share common properties and/or play similar roles within the graph»[18].

Now we have defined the concept of nodes, link, and groups, we deepen the foundation of the graph theory.

Adjacency Matrix

Given a network with N nodes, the matrix used to represent the connections between them is called *adjacency matrix* W, which is a $N \times N$ matrix. We label the N nodes with integers from 1 to N, thus we follow this rule to fill its entries:

$$W_{ij} = \begin{cases} 1 & \text{If a link exists between node } i \text{ and node } j \\ 0 & \text{Otherwise} \end{cases} \tag{3.1}$$

This the simplest way to construct an adjacency matrix. In case more complex networks, like *weighted network*, where the links have different weights or different relevance, the Eq. 3.2 becomes:

$$W_{ij} = \begin{cases} k & \text{If a link with weight } k \text{ exists between node } i \text{ and node } j \\ 0 & \text{Otherwise} \end{cases} \tag{3.2}$$

In the case of multiedges networks, the value k could be considered as the sum of weighted or unweighted links between the node i and the node j.

In many different networks, we observe symmetric relations between the elements: thus, the link between two nodes represents a bidirectional relation. This fact is easily represented in the adjacency matrix, since it is related to a symmetric adjacency matrix. Thus, we observe that $W = W^T$. This kind of networks are called *undirected networks*.

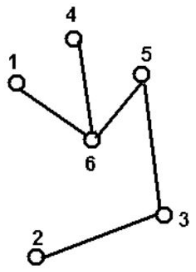

Figure 3.1: Example of undirected network

In the case of non symmetric relations, the adjacency matrix is not symmetric, i.e. $W \neq W^T$. These netowrks are called *directed networks* and the links are draws with an arrow to explicit the direction of the connection.

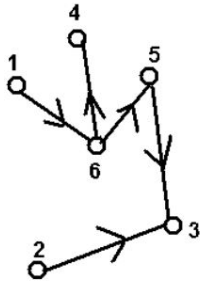

Figure 3.2: Example of directed network

Degree

The *degree* k_i of a node i is the sum of the edges connected to it. In the case of directed network, we define an *in-degree* and an *out-degree* as follows:

$$k_i^{in} = \sum_{j=1}^{N} W_{ij} \qquad k_j^{out} = \sum_{i=1}^{N} W_{ij} \tag{3.3}$$

In the case of undirected network, the degree in unique and defined as:

$$k_i = \sum_{j=1}^{N} W_{ij} \tag{3.4}$$

In undirected graphs, we observe that every link has two ends. Hence, in a graph with m egdes, we have $2m$ edges ends. But this number is equal to the sum of the degree of all the network, thus:

$$2m = \sum_{i=1}^{N} k_i \implies m = \frac{1}{2} \sum_{i=1}^{N} k_i \tag{3.5}$$

The mean degree c of the nodes of an undirected network is:

$$c = \frac{1}{N} \sum_{i=1}^{N} k_i \tag{3.6}$$

By using Eq. 3.5 we obtain:

$$c = \frac{2m}{n} \tag{3.7}$$

This result is useful to calculate the *density* ρ of the network. The density is the ratio between the existing links in the network and the number of the possible links:

$$\rho = \frac{m}{\binom{N}{2}} = \frac{2m}{N(N-1)} = \frac{c}{N-1} \tag{3.8}$$

This definition allow us to define *dense networks* and *sparse networks*. A network called dense if, while $N \to \infty$, ρ tends to a constant. On the contrary, a sparse network is defined by $\rho \to 0$ while $N \to \infty$.

Community

The concept of community is strictly related to real networks: if we observe their distribution of edges, we note that it is inhomogeneous, both globally and locally, with high concentrations of edges within special groups of vertices, and low concentrations between these groups [18]. This feature is called community structure. The analysis of community structures and their partition has been going on for years: the easiest operation to detect groups in a network consists of partitioning the vertices of a graph into clusters that are more densely connected [1]. In general, we note also that « community structures may also refer to groups of vertices that connect similarly to the rest of the graphs without having necessarily a higher inner density, such as disassortative communities that have higher external connectivity»[1].

In literature does not exist a definition of communities universally accepted: in fact, it is defined in the analyzed system case by case. Fortunato [18] analyzes different definitions of communities: local definitions (based on different criterion, like reachability or vertex degree, or by counting the internal number of edges between the nodes of the group respect to the rest of the network,), global definitions (all based on the idea the network is different from a random network defined as an Erdos-Renyi graph), and definition based on vertex similarity.

The most classical methods employed in communities clustering are graph partitioning, hierarchical clustering, partitional clustering, and spectral clustering. The last one will be further employed in our analysis. In general, many models consider a pre-establish structure in the network (like bipartite, assortative, or core-periphery), thus they define a cost function that is maximized or minimized to obtain the groups which best agree that structure. As we will study in the following, another approach is to define a network model that infer several properties from the network and thus to obtain the communities which best agree with these properties (like for the stochastic block model).

3.2 Model Definition

In this section, we define the stochastic block model, which belongs to the class of the random network. The base model for the random networks is the Erdos-Renyi model, which is defined

only with the probability p: this parameter expresses the probability to create an edge between each couple of nodes in the network. Hence, the probability of linking between all the nodes is the same, independently from the communities' structure. The stochastic block model generalizes the Erdos-Renyi model since the probability of linking two nodes depends on the group membership of each couple of nodes. The simplest version of the stochastic block model is described by:

- The scalar n, which is the number of vertices or nodes of the network;
- The scalar k, which is the number of communities or groups in the network;
- The vector $\boldsymbol{c}$, which relates the membership of each node to its group and we denote by c_i the group to which vertex i belongs;
- The affinity matrix $\boldsymbol{\Theta}$, which is a $k \times k$ matrix;

The affinity matrix entries describe the probability of connection between vertices, i.e. the probability to create an link between them:

$$\Theta = \begin{pmatrix} \theta_{11} & \theta_{12} & \cdots & \theta_{1k} \\ \theta_{21} & \theta_{22} & \cdots & \theta_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{k1} & \theta_{k2} & \cdots & \theta_{kk} \end{pmatrix} \tag{3.9}$$

Thus, the entries of $\boldsymbol{\theta}$, $\theta_{c_i c_j}$, are the independent probabilities to create an link between vertices i and j. We notice that the probability of connection depends only on the group memberships of the vertices.

3.3 Generative Model

In this section, we explore the stochastic block model as a generative model. First, we explain the terms "generative" and "inference". The stochastic block model defines a probability distribution over networks $Pr(G|\Phi)$. Φ is the set of parameters of the model and they are related to the probability of link creation. Knowing the values of Φ, by flipping a coin with biased (link) probability, we *generate* the network.
The other term refers to the reverse process: given a network G, we aim to obtain an estimate of the set of parameters Φ which give the highest probability to generate a network like G. This process is called *inference*.

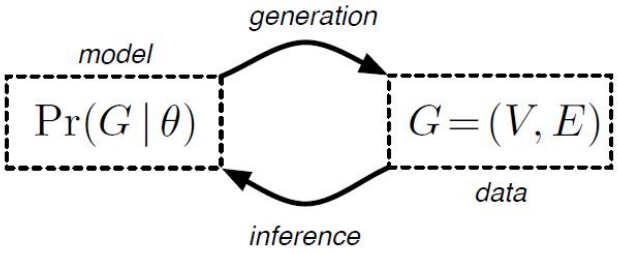

Figure 3.3: Generative model and inference process

Now we discuss the different kinds of network which this model can generate. The simple stochastic blockmodel is defined by a $k \times k$ matrix $\boldsymbol{\theta} = [\theta_{ab}]$, where k is the number of vertex classes in the network and θ_{ab} is the probability of creating an link between a node of the class a and a node of class b. The last vector of parameters is the vector $\boldsymbol{c}$, which is the vector of the class membership of the nodes. Thus, in the undirected case, we have to set the labeling of all the nodes and then specify the $\binom{k}{2}$ values in $\boldsymbol{\theta}$. To show some examples, we set the number of groups $k = 4$ and we fix the number of nodes, then we use different choices of $\boldsymbol{\theta}$. The easiest choice of $\boldsymbol{\theta}$ is $\theta_{ab} = \theta, \forall a, b \in k$.

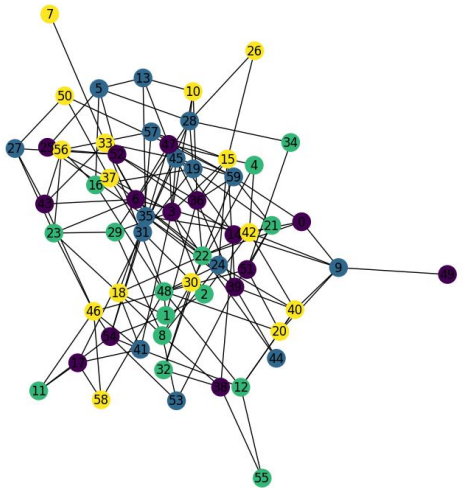

Figure 3.4: Erdos-Renyi network generated using the stochastic block model

In this case, the stochastic block model generates a pure random network. The model that describes a complete random network is the Erdos-Renyi model, and Fig. 3.4 shows an example of that kind of network.

A second choice of $\boldsymbol{\theta}$ is the following:

$$\Theta = \begin{pmatrix} k_1 & k_2 & k_2 & k_2 \\ k_2 & k_1 & k_2 & k_2 \\ k_2 & k_2 & k_1 & k_2 \\ k_2 & k_2 & k_2 & k_1 \end{pmatrix} \tag{3.10}$$

where $0 < k_1 < 1$ and $0 < k_2 < 1$. A similar choice of $\boldsymbol{\theta}$ can generate two kinds of network:

- for $k_2 < k_1$, we have more links between nodes of the same group: the structure of the generated network is called *assortative*;
- for $k_1 < k_2$, each node tends to be linked more with nodes of other groups: this kind of network structure is called *dissortative*;

An example of these two networks is shown in Fig. 3.5.

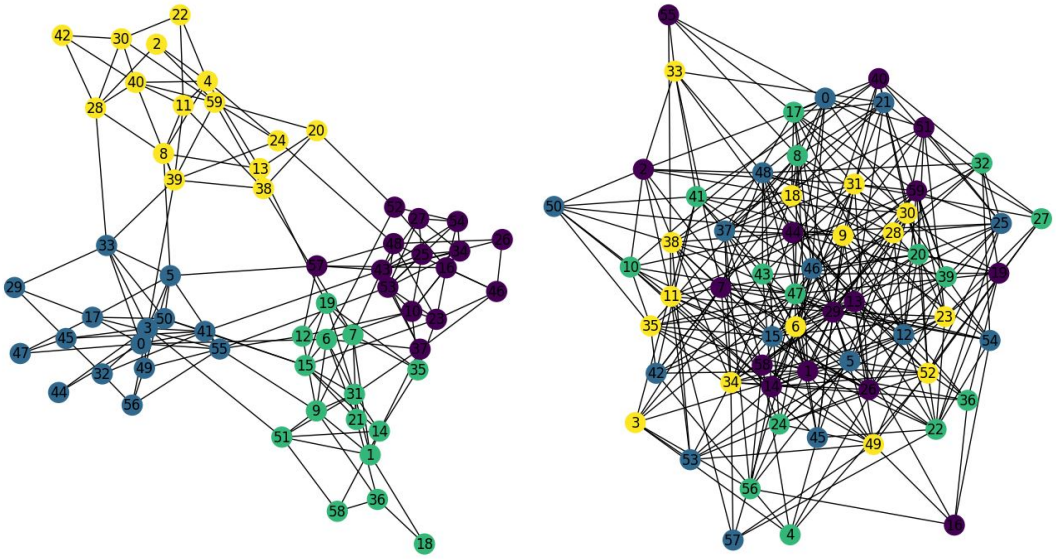

Figure 3.5: Example of assortative (on the left) and dissortative (on the right) network, generated by the exchange of k_1 and k_2

We can generate other two structures by changing entries in $\boldsymbol{\theta}$:

- We can consider a model where each node has a high probability of link creation between itself and the nodes of its same group or the nodes of the adjacent groups (we can numerically explicit the adjacent groups by numbering all the groups and consider adjacent a group if the difference between the group numbers is one). In this case, the

structure of Θ is:

$$\Theta = \begin{pmatrix} k_1 & k_2 & k_3 & k_4 \\ k_2 & k_1 & k_2 & k_3 \\ k_3 & k_2 & k_1 & k_2 \\ k_4 & k_3 & k_2 & k_1 \end{pmatrix} \tag{3.11}$$

where $0 < k_3 \leq k_2 \leq k_1 < 1$. The structure of this generated network is called *ordered* (see Fig. 3.6, left).

- We can consider a network where each group is more linked with its left-adjacent respect is right-adjacent (where left and right are considered respect the to ordinals) and each group has always less inner links respect its left-adjacent. An example of Θ for this network is:

$$\Theta = \begin{pmatrix} k_1 & k_2 & k_4 & k_6 \\ k_2 & k_3 & k_4 & k_6 \\ k_4 & k_4 & k_5 & k_6 \\ k_6 & k_6 & k_6 & k_7 \end{pmatrix} \tag{3.12}$$

Where $k_{i+1} < k_i < k_{i-1}$. This structure is called *core-periphery* (see Fig. 3.6, right).

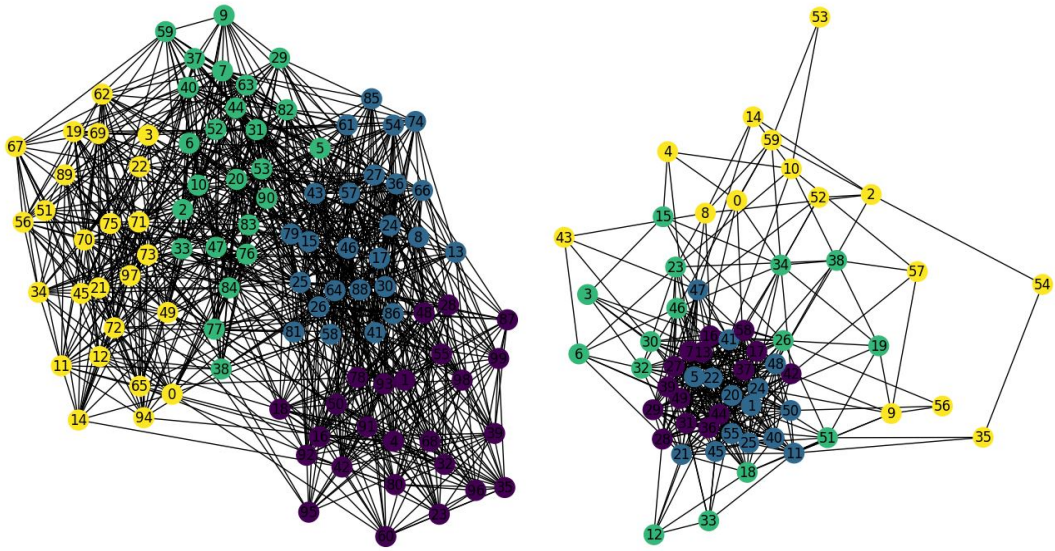

Figure 3.6: Example of ordered (on the left) and Core-Periphery (on the right) networks

By means of these simple examples, we have illustrated why the stochastic block model, in its simpler form, can be used to generate a large variety of networks with different community structures. Now we analysis the inference process.

3.4 Inference Process

Given an observed network G, the inference process gives an estimate of the parameters which describes the SBM with the highest probability to generate that observed network.
The inference process involving the stochastic block model assumes a given number of groups k. In order to infer the most likely k, different methods have been proposed, including Bayes factors, minimum description length approaches, Bayesian marginalization. [11]
Given the number of groups k, we can use different methods to estimate the best choice of the other parameters θ and $\boldsymbol{c}$, but one of the most employed is the maximum likelihood criteria [11] [40].

Let us consider an observed network G, and we identify with W its adjacency matrix. SBM is parameterized by a $k \times k$ matrix $\boldsymbol{\theta} = [\theta_{ab}]$, where θ_{ab} denotes the probability of forming an link between a node in class a and a node in class b. SBM decomposes the adjacency matrix into k^2 blocks, where each block is associated with relations between nodes in classes a and b. Each block (a, b) corresponds to a submatrix $W_{[a][b]}$ of the adjacency matrix W. Thus, given the class membership vector $\boldsymbol{c}$, each entry of W is an independent realization of a Bernoulli random variable with a block-dependent parameter, i.e., $w_{ij} \sim Bernoulli(\theta_{c_i c_j})$.
Since each entry of W is independent, the likelihood for the parameters Φ of the SBM is given by

$$P(W; \Phi) = \prod_{i \neq j} \left(\theta_{c_i c_j}\right)^{w_{ij}} \left(1 - \theta_{c_i c_j}\right)^{1 - w_{ij}} \tag{3.13}$$

The likelihood can be rewritten as:

$$P(W; \Phi) = exp\left\{ \sum_{a=1}^{k} \sum_{b=1}^{k} [m_{ab} \log \theta_{ab} + (n_{ab} - m_{ab}) \log (1 - \theta_{ab})] \right\} \tag{3.14}$$

where $m_{ab} = \sum_{i \in a} \sum_{j \in b} w_{ij}$ denotes the number of observed links in block (a, b), and

$$n_{ab} = \begin{cases} |a||b| & a \neq b \\ |a|(|a| - 1) & a = b \end{cases} \tag{3.15}$$

where $|a|$ denotes the number of nodes in the group a and n_{ab} denotes the number of possible links in block (a, b).
Based on the knowledge we have of G, the inference process can be named *a priori* if we know the labels $\boldsymbol{c}$ and we have to estimate only $\boldsymbol{\theta}$, or it can be called *a posteriori* if $\Phi = \{\boldsymbol{\theta}, \boldsymbol{c}\}$.

In the a priori setting, a sufficient statistic for estimating $\boldsymbol{\theta}$ is the matrix $\boldsymbol{Y}$ of the block densities corresponding to ratios of observed links relative to possible links within each block, which has entries

$$y_{ab} = \frac{m_{ab}}{n_{ab}} \tag{3.16}$$

The matrix $\boldsymbol{Y}$ is the maximum-likelihood estimate of $\boldsymbol{\theta}$.

Parameter estimation in the a posteriori setting is more involved, and many methods have been proposed, including Gibbs sampling [32], label-switching [29] [41], and spectral clustering [34] [37]. The label-switching methods use a heuristic for solving the combinatorial optimization problem of maximizing the likelihood over the set of possible class memberships, which is too large for an exhaustive search to be tractable. The spectral clustering methods utilize the eigenvectors of the adjacency matrix W to estimate the class memberships.

3.4.1 The spectral clustering method

Following the analysis of Sussman at al. [37], we briefly analyze this clustering method. The spectral clustering method is one of the possible algorithms to assign nodes to their correct blocks. The main steps of the algorithm are:

- We calculate the singular value decomposition of the adjacency matrix $W = \tilde{U}'\tilde{\Sigma}'\tilde{V}'^{T}$. Let $\tilde{\Sigma}'$ have decreasing entries on its main diagonal;
- We define $\tilde{U}$ and $\tilde{V}$ as the first d columns of $\tilde{U}'$ and $\tilde{V}'$, and $\tilde{\Sigma}$ will be the sub-matrix of $\tilde{\Sigma}'$ given by the first d columns and rows;
- $Z \leftarrow [\tilde{U}\tilde{\Sigma}^{1/2}, \tilde{V}\tilde{\Sigma}^{1/2}]$, which is the concatenation of the coordinate-scaled singular vector matrices;
- We compute $\boldsymbol{c}$ using the *k-means* clustering on rows of Z;

This algorithm computes the singular value decomposition of W, reducing the dimension. We replace the last instruction of the algorithm with the one used by Xu and Hero. The original one is:

- We define $(\hat{\psi}, \hat{\tau}) = argmin_{\psi,\tau} \sum_{u=1}^{n} ||\tilde{Z}_u - \psi_{\tau(u)}||_2^2$ the function which give the centroids and the block assignments, where $\tilde{Z}_u$ is the u^{th} row of $\tilde{Z}$, $\hat{\psi} \in \Re^{K\times d}$ are the centroids and $\hat{\tau}$ is a function from $\boldsymbol{c}$ to $[k]$, where $[k]$ is a vector and its entries are the numbering of the groups in the network;

In fact, « this procedure it's a mathematically convenient stand-in for what might be used in practice» [37]. *K-means* represents the empirical evaluation of that mathematical expression to minimize the square error.

3.4.2 The detectability problem

Now we are introducing one of the problems in SBM, called the detectability-undetectability problem. We have to introduce the planted partition model to proceed in our analysis, since is the model used by Decelle et al. [13] to study the detectability-undetectability problem. The planted partition model is a particular parametrization of the stochastic block model where the affinity matrix is filled with only two different values: c_{out} and c_{in}. The structure of the affinity matrix can be both assortative and disassortative. This parametrization is a benchmark for the stochastic block model[13][30] and it is employed to study the groups' partitioning problem.
Let us now consider a network G generated as explained in Section 3.3 and let N_a be the number of nodes in the group a. Since N_a is binomially distributed, in the limit $N \to \infty$, each N_a is concentrated around its mean $g_a N$, where g_a is the expected fraction of nodes in the group a. The average number of links from the group a to group b is then $m_{ab} = p_{ab} N_a N_b$, or $m_{aa} = p_{aa} N_a (N_a - 1)$ if $a = b$. If we consider a sparse graph, where $p_{ab} = O(1/N)$, it is useful to work with the rescaled affinity matrix $c_{ab} = N p_{ab}$. In the limit of large N, the average degree is then:

$$c = \sum_{a,b} c_{ab} g_a g_b \tag{3.17}$$

In the undirected case W_{ij}, p_{ab}, and c_{ab} are symmetric and the average degree is:

$$c = \sum_{a<b} c_{ab} g_a g_b + \sum_{a} c_{aa} \frac{g_a^2}{2} \tag{3.18}$$

Since we have defined c, we can also define the parameter ϵ, which is a common parameter for the planted partition model:

$$\epsilon = \frac{c_{out}}{c_{in}} \tag{3.19}$$

This parameter is defined to be:

$$\epsilon = \begin{cases} 1 & \text{Erdos-Renyi random graph} \\ 0 & \text{Groups which are completely separated} \end{cases} \tag{3.20}$$

We can define a measure of correct inference of the group assignment, called *agreement*:

$$A(\{t_i\}, \{q_i\}) = max_\pi \frac{1}{N} \sum_i \delta_{t_i, \pi(q_i)} \tag{3.21}$$

where t_i and q_i are the original assignment of the nodes of the generated network G and its estimate, respectively, and π ranges over the permutations on q elements. Decelle et al [13]. define a normalized agreement, called overlap:

$$Q(\{t_i\}, \{q_i\}) = max_\pi \frac{\frac{1}{N} \sum_i \delta_{t_i, \pi(q_i)} - max_a g_a}{1 - max_a g_a} \tag{3.22}$$

This measure is defined so that if we find the exact labeling of the nodes, then $Q = 1$ and zero if labeling is random.

We have defined ϵ and overlap in order to analyze the results obtained Decelle et al in this paper [13]: in fact, in Fig. 3.7 [13], we observe the relation between the overlap and the parameter ϵ both for large graphs and both for graphs with a small number of nodes: on the left the simulated graphs are divided into 2 groups, on the right the graphs are divided into 4 groups.

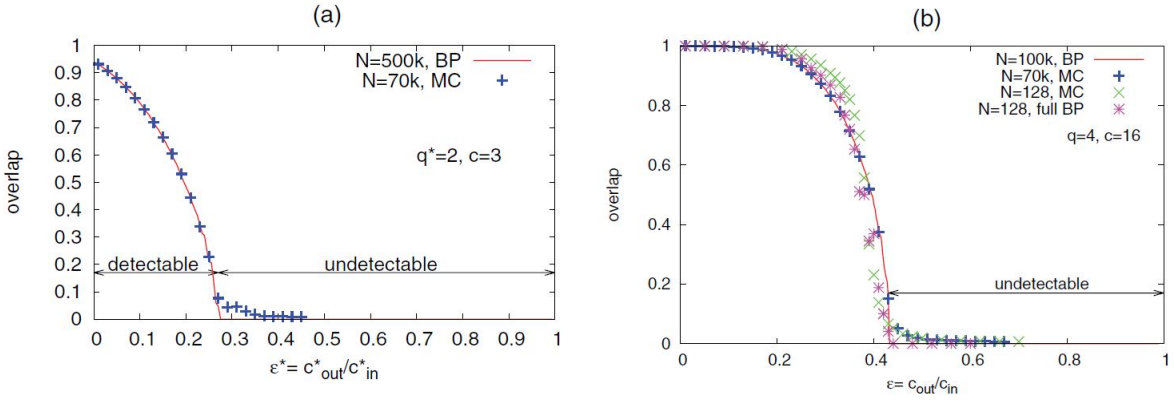

Figure 3.7: Overlap for network with 2 and 4 groups varying the parameter ϵ^*[13] The graph compares the belief propagation results (red line) to MCMC (Monte Carlo Markov Chain) simulations (data point): on the left we observe data from simulations for large graphs, while on the right the graphs contain a small number of nodes

Decelle et al. [13] generate many networks by varying the value of ϵ and they try to recover the network's structures through an algorithm which combines the cavity equation with the believe propagation (BP), by maximization of the likelihood of the model. Decelle et al. observe that, both for large graphs and for small graphs, if the parameter ϵ is greater than a certain threshold, the structure of the graphs is undetectable: they demonstrates that [13], « for $N \to \infty$ and $\epsilon > \epsilon_c = (c - \sqrt{c})/[c + \sqrt{c}(k-1)]$ it is impossible to find an assignment correlated with the original one based purely on the structure of the graph».
Thus, we observe that, independently from the algorithm chosen to infer the network structure, it is not always possible to recover the exact labeling of the nodes in a network using only its structure.

3.5 Conclusion

The stochastic block model is nowadays one of the benchmark models to generate networks with a community structure and to infer the parameters of an observed network to learn information of real-world systems. As we observed in this chapter, SBM could be employed to generate networks with very different structures simply modifying the entries of the affinity matrix Θ.

Furthermore, we have many algorithms and methods to infer the structure of an observed graph and they are based on the SBM and its generalizations. However, the model studied in this chapter is useful only for non time-dependent graphs, also referred as *static* networks.
In the following, we analyze a specific dynamic generalization of the SBM, proposed by Xu and Hero in 2013 [40]. Then, in the subsequent chapter, we implement our version of the estimation method proposed by the authors, for a comparative analysis: we point out some critical elements, thus proposing novel models and a generalized estimation method to overtake some drawbacks for applications.

Chapter 4

Dynamic Stochastic Block Model: Analysis and Numerical Simulations

In the previous chapter, we analyzed the stochastic block model and how this model could be employed to generate networks or inferring data from graphs. This model is used for non time-evolving networks and for this reason, is also called *static* stochastic block model.
In this chapter, we analyze one of the many models which describe time-evolving networks, also referred as *dynamic* networks. Our analysis is mainly based on the dynamic stochastic block model proposed by Xu and Hero [40], which is the baseline model for some generalization proposed in the following.
This chapter is divided into three sections:

- Analysis of the authors' model;
- Numerical simulations' settings;
- Analysis of the numerical simulations' results, under different model specifications;

The first section is divided into two subsections, where we review the authors' generative model and the inference process. The notation will be the same as in the previous chapter, we add an apex "^{t}" to denote the time step we are analyzing. This section is mainly based on the article [40].
The second and third sections are original. The second section specifies the code used in the numerical simulations and the settings of the model's parameters. In the third section we study the results of the numerical simulations of the model. We divide this section into four subsections:

1. Study of the generative model for long-time simulations;
2. Low-density regime: analysis of the inference process depending on the number of nodes, keeping fixed the number of groups;

3. Low-density regime: analysis of the inference process depending on the number of groups, keeping fixed the number of nodes;

4. Analysis of the inference process depending on the fraction of nodes that change classes between time steps;

Regarding the simulations of the Xu and Hero model, we consider the same metrics adopted in [40] for a fair comparison between the results of the paper and the ones obtained by our implementation of the method. In particular, we point out that it is crucial considering the presence of the observation noise, thus using an estimation method based on the extended Kalman filter, for networks which are both small (in the number of nodes) and sparse (in the number of links).

4.1 Xu and Hero Dynamic SBM

In this section, we examine the dynamic stochastic block model proposed by Xu and Hero. The authors propose a vectorization of the parameter of the SBM: the columns of the affinity matrix Θ will be stacked on top of each other, and the same for the other parameters. In the inference process, we study both the analytical model and the pseudo-code proposed by the authors. The latter is the benchmark for the code used in our numerical simulations.

4.1.1 Generative model

The generative model for the dynamic network is more complex than a static network generative model: for SSBM (static stochastic block model), we just need to flip a biased coin for each couple of nodes to create the adjacency matrix. For a dynamic network, we want a time-dependence between a snapshot and the next one.

Xu and Hero propose the following process to describe the evolution of the affinity matrix determining the link probability between groups or within the same group:

$$\boldsymbol{\psi}^t = F^t \boldsymbol{\psi}^{t-1} + \boldsymbol{v}^t \tag{4.1}$$

where:

- $\boldsymbol{\psi}^t$ is the vector representation of the matrix Ψ^t, defined as:

$$\Psi^t = [\log \theta^t_{ab} - \log (1 - \theta^t_{ab})] \in \mathfrak{R}^{k \times k} \tag{4.2}$$

- F^t is the state transition matrix applied to the previous state and set equal to the Identity matrix by the authors;

- $\boldsymbol{v}^t$ is a random vector of zero-mean Gaussian entries with covariance matrix Γ^t.

The random vector $\boldsymbol{v}^t$ is also called *the process noise*.
Once obtained the values of $\boldsymbol{\psi}^t$, by applying the logistic function to $\boldsymbol{\psi}^t$, we obtain the values of θ^t_{ab}, the affinity matrix at the time step t. We need another parameter to generate the network at time t: the group memberships of the nodes $\boldsymbol{c}$. There exist many ways to describe how the community structure of a network changes over time. We follow the choice by Xu and Hero of reassigning a fixed percentage of nodes to other groups at each time step.

4.1.2 Inference process

The inference process is based on some assumption on the network G^t:

- The entries of $W^t_{[a][b]}$ are independent and identically distributed Bernoulli random variables (θ^t_{ab}). That means the sample mean y^t_{ab}, defined in Eq. 3.16, follows a rescaled binomial distribution. Thus, y^t_{ab} is approximately Gaussian by the Central Limit Theorem with mean θ^t_{ab} and variance:

$$(\sigma^t_{ab})^2 = \theta^t_{ab}(1 - \theta^t_{ab})/n^t_{ab} \tag{4.3}$$

  where n^t_{ab} is defined in equation 3.15.

- We assume that y^t_{ab} is Gaussian for all (a, b);

- The initial state is Gaussian distributed: $\boldsymbol{\psi}^0 = N(\boldsymbol{\mu}^0, \Gamma^0)$

Thus, Xu and Hero propose this linear observation model:

$$Y^t = \Theta^t + Z^t \tag{4.4}$$

where Z^t is a zero-mean independent Gaussian noise matrix with variance $(\sigma^t_{ab})^2$ for the $(a, b)th$ entry. Y^t and Θ^t are defined in Chapter 3 in Eq. 3.9 and Eq. 3.16. The authors propose a model where the observations of the network to be inferred is the complete set of the adjacency matrices, i.e. the observations would be given by the set $Y^{(t)}$, where $Y^{(t)} = \{Y^1, \ldots, Y^t\}$.

If we consider the generative model in Subsection 4.1.1, the observation model can be written in terms of $\boldsymbol{\psi}^t$ as:

$$\boldsymbol{y}^t = h(\boldsymbol{\psi}^t) + \boldsymbol{z}^t \tag{4.5}$$

where the function $h(x)$ is the logistic function applied to each entry of $\boldsymbol{\psi}^t$. We also denote the covariance matrix of $\boldsymbol{z}^t$ by Σ^t, which is a diagonal matrix with entries $(\sigma^t_{ab})^2$.

The inference process of the parameters of the network G^t is done using only the data from time t and earlier. We assume that $\{\boldsymbol{\psi}^0, \boldsymbol{v}^1, \ldots, \boldsymbol{v}^t, \boldsymbol{z}^1, \ldots, \boldsymbol{z}^t\}$ are mutually independent.

Since the observation process 4.5 is not linear, we apply the Extended Kalman Filter (see Chapter 2) to linearize the dynamics. This choice gives us the «near-optimal estimate of $\boldsymbol{\psi}^t$» [40]. The predicted state estimate is:

$$\hat{\boldsymbol{\psi}}^{t|t-1} = F^t \hat{\boldsymbol{\psi}}^{t-1|t-1} \tag{4.6}$$

and the predicted covariance estimate is:

$$\hat{R}^{t|t-1} = F^t \hat{R}^{t-1|t-1} (F^t)^T + \Gamma^t \tag{4.7}$$

The Kalman gain is given by:

$$K^t = \hat{R}^{t|t-1} (H^t)^T \left[H^t \hat{R}^{t|t-1} (H^t)^T + \Sigma^t \right]^{-1} \tag{4.8}$$

where H^t is a diagonal matrix and its entries are the derivative of the logistic function applied to $\boldsymbol{\psi}^{t|t-1}$. The update state estimate is:

$$\hat{\boldsymbol{\psi}}^{t|t} = \hat{\boldsymbol{\psi}}^{t|t-1} + K^t \left[\boldsymbol{y}^t - h(\hat{\boldsymbol{\psi}}^{t|t-1}) \right] \tag{4.9}$$

and the updated covariance estimate:

$$R^{t|t} = (I - K^t H^t) R^{t|t-1} \tag{4.10}$$

In the case of *a posteriori* inference process, the vector $\boldsymbol{c}^t$ is not known and must be estimated along with Ψ^t. Xu and Hero maximize the posterior state density given the entire sequence of observations $W^{(t)}$: this is obtained by alternating between label-switching and applying the EKF to obtain a maximum a posteriori probability (MAP).
The posterior state density is given by:

$$f(\boldsymbol{\psi}^t | W^{(t)}) \propto f(W^{(t)} | \boldsymbol{\psi}^t W^{(t-1)}) f(\boldsymbol{\psi}^t | W^{(t-1)}) \tag{4.11}$$

Thanks to the conditional independence of current and past observations given the current state $W^{(t-1)}$, we drops out the first multiplicative factor on the right side 4.11. This factor can thus be obtained simply substituting $h(\Psi^t)$ for Θ^t in Eq. 3.14. We approximate the second term in Eq. 4.11 with $f(\boldsymbol{\psi}^t | y^{(t-1)})$ using the estimated class memberships at all previous time steps. By applying the Kalman filter to the linearized temporal model, it is $f(\boldsymbol{\psi}^t | y^{(t-1)}) \sim N(\hat{\boldsymbol{\psi}^{t|t-1}}, R^{t|t-1})$. Thus, the logarithm of the posterior density is given by:

$$p^t = \log f(W^{(t)} | \boldsymbol{\psi}^t) + \log f(\boldsymbol{\psi}^t | y^{(t-1)}) \tag{4.12}$$

where:

$$\log f(W^{(t)} | \boldsymbol{\psi}^t) = \sum_{a=1}^{k} \sum_{b=1}^{k} \left\{ m_{ab}^t \log h(\psi_{ab}^t) + (n_{ab}^t - m_{ab}^t) \log \left[1 - h(\psi_{ab}^t) \right] \right\} \tag{4.13}$$

$$\log f(\boldsymbol{\psi}^t | y^{(t-1)}) = -\frac{1}{2}\left(\boldsymbol{\psi}^t - \boldsymbol{\psi}^{t|t-1}\right)^T \left(R^{t|t-1}\right)^{-1} \left(\boldsymbol{\psi}^t - \boldsymbol{\psi}^{t|t-1}\right) \tag{4.14}$$

We use the log-posterior as the objective function in the label-switching method.
To infer the correct labels of the nodes, we use a simple local search algorithm (hill-climbing) initialized using the estimated class memberships at the previous time step. This assumption can work only if the number of nodes that change groups between time steps is a small fraction of the total number.

The pseudo-code used to infer the parameters of the model of network is:

1. $\hat{\boldsymbol{c}}^t \leftarrow \hat{\boldsymbol{c}}^{t-1}$, which initializes the class membership
2. Compute block densities $\boldsymbol{Y}^t$ using W^t and $\hat{\boldsymbol{c}}^t$
3. Compute $\boldsymbol{\psi}^t$ using the EKF equations
4. Compute the log-posterior p^t by substituting $\boldsymbol{\psi}^{t|t}$ for $\boldsymbol{\psi}^t$ in 4.12
5. while $(iter < max)$ do:
6. $\quad \bar{p}^t \leftarrow -\infty$
7. $\quad \tilde{\boldsymbol{c}}^t \leftarrow \hat{\boldsymbol{c}}^t$
8. $\quad$ **for** $i = 1$ **to** $|V^t|$ **do**:
9. $\quad\quad$ **for** $j = 1$ **to** k such that $\hat{c}_i{}^t \neq j$ **do**:
10. $\quad\quad\quad \tilde{c}_i{}^t \leftarrow j$, so we change the class of a single node
11. $\quad\quad\quad$ Compute block densities $\tilde{\boldsymbol{Y}}^t$ using W^t and $\tilde{\boldsymbol{c}}^t$
12. $\quad\quad\quad$ Compute $\tilde{\boldsymbol{\psi}}^t$ using the EKF equations
13. $\quad\quad\quad$ Compute the log-posterior $\tilde{p}^t$ by substituting $\boldsymbol{\psi}^{t|t}$ for $\boldsymbol{\psi}^t$ in 4.12
14. $\quad\quad\quad$ **if** $\tilde{p}^t > \hat{p}^t$ **then**
15. $\quad\quad\quad\quad [\bar{p}^t, \bar{\boldsymbol{\psi}}^t, \bar{\boldsymbol{c}}^t] \leftarrow [\tilde{p}^t, \tilde{\boldsymbol{\psi}}^t, \tilde{\boldsymbol{c}}^t]$
16. $\quad\quad\quad \tilde{c}_i{}^t \leftarrow \hat{c}_i{}^t$
17. $\quad\quad$ **if** $\bar{p}^t > p^t$ **then**:
18. $\quad\quad\quad [p^t, \hat{\boldsymbol{\psi}}^{t|t}, \hat{\boldsymbol{c}}^t] \leftarrow [\bar{p}^t, \bar{\boldsymbol{\psi}}^t, \bar{\boldsymbol{c}}^t]$
19. $\quad\quad$ **else break**

20. **return** $[\hat{\boldsymbol{\psi}}^{t|t}, \hat{\boldsymbol{c}}^{t}]$

where the procedures from 9 to 18 explicit the combination of label-switching (procedures 9-10) and log- posterior maximization (procedure 13-14-15-16-17). The idea behind the label switching algorithm is to change the label for a node in the network and then recalculate the log-posterior in that configuration. If we obtain better result respect to the previous one, the label of the nodes is changed. These few instructions are repeated for each node at each time step.

The inference procedure algorithm depends on four hyperparameters:

- The mean $\boldsymbol{\mu}^1$ of the initial state $\boldsymbol{\psi}^1$;
- The covariance matrix Γ^1 of the initial state $\boldsymbol{\psi}^1$;
- The covariance matrix Σ^t of the observation noise $\boldsymbol{z}^t$;
- The covariance matrix Γ^t of the process noise $\boldsymbol{v}^t$.

The first two hyperparameters are related to the initial state. In the absence of prior information about the network, specifically the matrix Θ^1 of link probabilities, we let the variances of the initial states approach zero.
The third hyperparameter Σ^t denotes the covariance matrix of the observation noise. In many applications of state-space models, it is assumed to be time-invariant and estimated jointly with Γ^t. In the dynamic SBM setting, Σ^t is assumed to be time-varying and it is related to the current state $\boldsymbol{\psi}^t$ through Eq. 4.3 and the logistic function. Thus, we use a plug-in estimator for Σ^t by substituting $h(\boldsymbol{\psi}^{t|t-1})$ in Eq. 4.3.
The final hyperparameter Γ^t denotes the covariance matrix of the process noise $\boldsymbol{v}^t$. Unlike Σ^t, we assume Γ^t to be time-invariant. Furthermore, it is not necessarily diagonal because of states could evolve in a correlated manner. For example, if $\boldsymbol{\psi}^t_{ab}$ increases from time $t-1$ to time t, it may be a result of some action by nodes in class a, which could also affect the other entries in a row a of Ψ^t. Although Γ^t is not necessarily diagonal, it is desirable to impose some structure on Γ^t to reduce its dimensionality. Hence, Xu and Hero assume the following generalization for Γ^t:

$$\gamma^t_{ij} = \begin{cases} s_{diag}, & i = j \\ s_{nb}, & i, j \text{ are neighboring cells in } \Psi^t \\ 0, & \text{otherwise} \end{cases} \tag{4.15}$$

where i, j being neighboring cells means that the matrix indices (a_i, b_i) corresponding to i in Ψ^t are in the same row or column as matrix indices (a_j, b_j).

4.1.3 Model analysis

The dynamic stochastic block model described by Xu and Hero is one of the possible models to infer and generate dynamic networks. We summarize here its main features:

- The evolution model described in Eq. 4.1 is mathematically easy to handle because it depends on two hyperparameters: the state evolution matrix F^t and covariance matrix Γ^t which generates the process noise v^t;
- The assignments of the nodes between time steps have only one limitation: the fraction of nodes that change group must be small.

Furthermore, Xu and Hero impose that the evolution matrix F^t is fixed in time and it is equal to the identity matrix: thus, the evolution model becomes:

$$\boldsymbol{\psi}^t = \boldsymbol{\psi}^{t-1} + \boldsymbol{v}^t \tag{4.16}$$

This equation describes a random walk of the logistic function of the affinity matrix entries. Hence, each entry of $\boldsymbol{\psi}^t$ tends to large values for long time generations, if the multiplication of the time for the process variance is large enough. This fact implies that, by considering the logistic function which relates $\boldsymbol{\psi}^t$ to $\boldsymbol{\theta}^t$, the affinity matrix entries are likely to approach zero or one. This means that some groups in the network are completely linked and others have not a single link between them. We can say that this generative model creates *absorbing states* if the generation process is repeated for enough time.

4.2 Numerical Simulations' Settings

In this section, we examine the code used to generate dynamic networks and to infer their parameters. We can divide this section into three parts:

- Analysis of the code;
- Settings of the code to generate networks and to infer the parameters;
- Definition of the measures (ARI, MSE, MSPE) used to test the goodness of the inference process;

4.2.1 Analysis of the code

The code employed in this thesis is original and it combines the functions of standard Python libraries (*random, numpy, math, time, matplotlib.pyplot, networkx, sklearn.metrics and sklearn.cluster* with an original Python class called *Rete*. The class contains a list of functions and members that we can group into:

- Constructors;
- Matrices or vectors generators or populators;

- Functions which make evolve the network through the time;
- Specific algorithm (EKF, Spectral Clustering, posterior likelihood estimation...)

These members can be mixed to create both a generative model and an inference process.

A generic simulation is so structured:

1. **Network definition**: in this part, we define the number of nodes N in the network, the number of groups k, how the nodes are divided among the groups, the simulation time T, the initial condition of the Eq. 4.1, and the fraction of nodes that change classes between time steps;
2. **Network generation**: in this part, we create the dynamic network by making evolve the network created at time $t = 0$; here we define the matrix Γ^t and we construct v^t;
3. **Estimate of the model's hyperparameters**;
4. **Estimate of network's parameters**: in this part, we convert the pseudo-code proposed by Xu and Hero 4.1.2 into real code;
5. **Plot generation and comparison**.

The code is written to use a single-core, but it could be parallelized to speed up the execution of the process.

4.2.2 Parameters setting

We first set the parameters used in Xu and Hero [40] to simulate the network.
We set the number of nodes equal to 128, divided in 4 groups. The affinity matrix of the initial state is the one of a planted partition model, specifically:

$$\theta^0 = \begin{pmatrix} 0.2580 & 0.0836 & 0.0836 & 0.0836 \\ 0.0836 & 0.2580 & 0.0836 & 0.0836 \\ 0.0836 & 0.0836 & 0.2580 & 0.0836 \\ 0.0836 & 0.0836 & 0.0836 & 0.2580 \end{pmatrix}$$

The initial state ψ^0 is generated with a normal distribution whose mean is θ^0 and the covariance matrix is $\Gamma^0 = 0.04I$. ψ^0 represents the state at time 0 and it is used to generate the state at time 1, i.e. ψ^1, through Eq. 5.1.
The covariance matrix of the process noise, Γ^t, is static and, according to the parametrization in Eq. 4.15, it is:

$$\begin{cases} s_{diag} = 0.01 \\ s_{nb} = 0.0025 \end{cases} \tag{4.17}$$

The labels of the nodes change through time. in this case, every time step 10% of nodes randomly is moved from a group to another. Given the node labels and the state variables, the network W^1 is generated.
To test the algorithm, 10-time steps are generated.

4.2.3 Inference process and the metrics

In simulation studies, once a sample of networks is produced according to an underlying model, the problem is to infer the model parameters starting from the simulated data and the goodness of fit of the adopted inference method. Regarding the Xu and Hero model, the goal is the inference of the four hyperparameters, as described in Chapter 4. Here, we assume to know the initial mean and covariance of the state 0. At time t = 1, in absence of prior information about the network, we employ a diffuse prior [15]; that is, we let the variances of the initial states to be equal to zero. This can be implemented by simply taking $\psi^{1|1} = g(y^1)$ and $R^{1|1} = G^1\Sigma^1(G^1)^T$, where $g_i(\boldsymbol{x}) = h_i^{-1}(x) = log(x_i) - log(1 - x_i)$ is the logit of the th entry of x, and G^1 is the Jacobian of g evaluated at y^1, which is a diagonal matrix with entries given by $g_i'(y1) = 1/y_i^1 + 1/(1 - y_i^1)$. Thus, the initial state mean and covariance are given by the transformed initial observation and its covariance. The third hyperparameter Σ^t denotes the covariance matrix of the observation noise z^t. In the dynamic SBM setting, however, Σ^t is estimated to be time-varying, in fact, it is related to the current state Ψ^t through the logistic function $h(x)$. We use a plug-in estimator for Σ^t by substituting $h(\hat{\psi}^{t|t1})$ in the second equation of 5.1. The final hyperparameter Γ^t is the covariance matrix of the process noise v^t and it is estimated by exploring the domain and minimizing the Mean Square Prediction Error (MSPE).

To test the quality of this model, we use two metrics: the Mean Square Error (MSE) and the Adjusted Rand Index (ARI). The MSE is calculated as the Mean Square Tracking Error $||\psi^{t|t} - \psi^t||_2^2$ over the 10-time steps.
The Adjusted Rand Index in statistics is a measure of the similarity between two data clustering [33].
Given a set S of n elements, and two groupings or partitions of these elements, namely $X = \{X_1, X_2, \ldots, X_r\}$ and $Y = \{Y_1, Y_2, \ldots, Y_s\}$, the overlap between X and Y can be summarized in a contingency table $\left[n_{ij}\right]$, where each entry n_{ij} denotes the number of objects in common between X_i and Y_j : $n_{ij} = |X_i \cap Y_j|$.
The ARI is defined as:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}]/\binom{n}{2}} \tag{4.18}$$

where a, b, n_{ij} are:

$X \diagdown Y$	Y_1	Y_2	...	Y_s	Sums
X_1	n_{11}	n_{12}	...	n_{1s}	a_1
X_2	n_{21}	n_{22}	...	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}	...	n_{rs}	a_r
Sums	b_1	b_2	...	b_s	

Thus, the MSE is used to measure the distance between the predicted $\psi^{t|t}$ and the true ψ^t, while ARI is used to measure the number of correctly inferred labels.

4.3 Analysis of the Numerical Results under Different Model Specifications

The Xu and Hero model, using the parameters specified by the authors, obtains an excellent score in both the metrics we used (and also the authors used to validate their model) [40].
In this section we will change these parameters to investigate deeply the model:

1. We use the Xu and Hero model as a generative model for a temporal network and we analyze its behavior when the model generates a network with a large number of time steps;

2. We fix the number of groups of nodes in the network and we vary the density of nodes per group (i.e. the number of nodes in each group) by varying the number of nodes in the network;

3. We fix the number of nodes in the network and we vary the density of nodes per group by varying the number of group in the network;

4. We vary the fraction of nodes that change classes between time steps in a large range.

4.3.1 Generative model: long-time simulations and absorbing states

The number of time steps used by Xu and Hero is 10. However many temporal series collect hundreds of time steps. So it could be interesting to investigate what happens if we generate a temporal network with the Xu and Hero model with a larger number of time steps.
The main result is that the evolution of the network is largely determined by the parameter Γ^t.
We know that the coefficients of the affinity matrix θ^t are related to ψ^t through the equation:

$$\theta^t_{i,j} = \frac{1}{1 + e^{-\psi^t_{i,j}}} \tag{4.19}$$

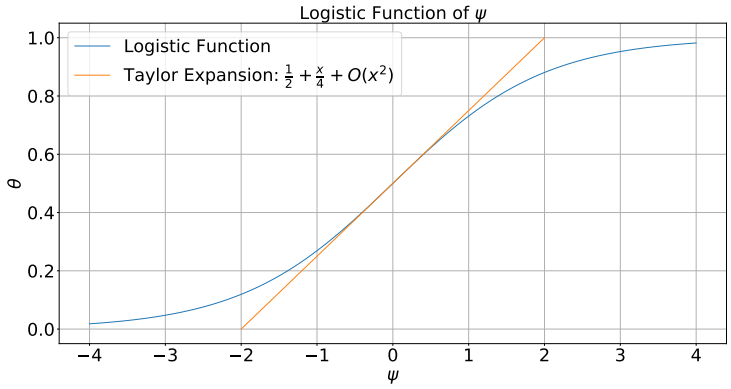

Figure 4.1: Logistic Function of ψ and its Taylor expansion

From Eq. 4.1, we know that $\boldsymbol{\psi}^t$ performs a random walk and the random walk depends on the vector v^t. Now we explore what happens to the affinity matrix Θ^t by varying the matrix Γ^t, which is used to generate the vector v^t, in a generated network with 1000 time steps.

First, we analyze the relation between θ^t and ψ^t. For $\boldsymbol{\psi}^t$ in range $[-1, 1]$, the fluctuations are linearly dependent on the entries of θ^t in the range $[0.27 - 0.73]$. In fact, the Taylor expansion of the logistic function in Eq. 4.19:

$$\theta^t_{i,j} = \frac{1}{2} + \frac{\psi^t_{i,j}}{4} - \frac{(\psi^t_{i,j})^3}{48} + O((\psi^t_{i,j})^4) \tag{4.20}$$

as a consequence, the Taylor expansion of θ^t results linear in ψ^t in range $[-1, 1]$, as we can see in Fig. 4.1. Also in the range $[-4, -1] \vee [1, 4]$, the variation in θ^t is still appreciable but, for ψ^t in the range $[-\infty, -4] \vee [4, \infty]$, the variation in θ^t is very small and its value becomes really close to 0 or 1. Thus, the affinity matrix is quite constant in time and it is really close to a matrix of 0 and 1.

We set the values ± 4 as threshold values in simulations, because for those values of $\boldsymbol{\psi}^t$:

$$\theta^t_{ij} = \frac{1}{1 + e^{-4}} = 0.018$$

$$\theta^t_{ij} = \frac{1}{1 + e^{4}} = 0.982$$

In the following Fig. 4.2, 4.3, 4.4 and 4.5, we can observe the effect on the affinity matrix θ, while increasing the entries of the covariance matrix Γ^t:

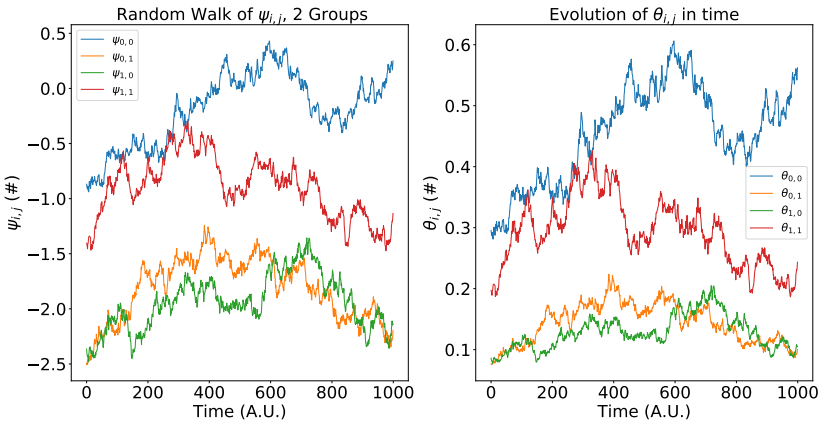

Figure 4.2: Evolution in time of ψ and θ, $s_{diag} = 0.001$ and $s_{snb} = 0.00025$

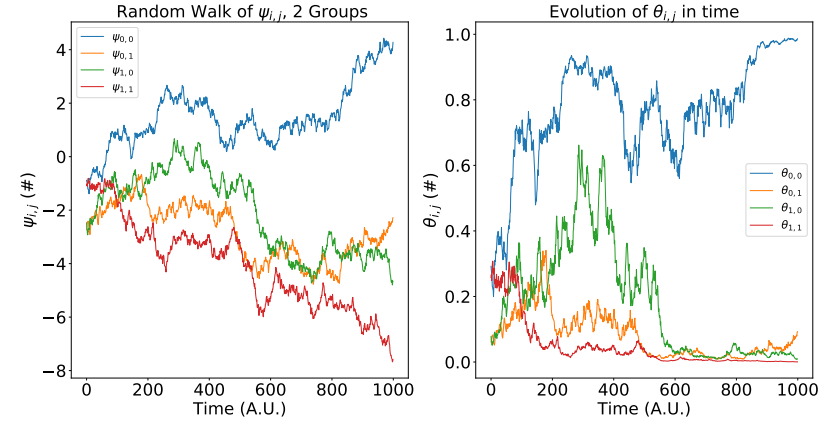

Figure 4.3: Evolution in time of ψ and θ, $s_{diag} = 0.01$ and $s_{snb} = 0.0025$

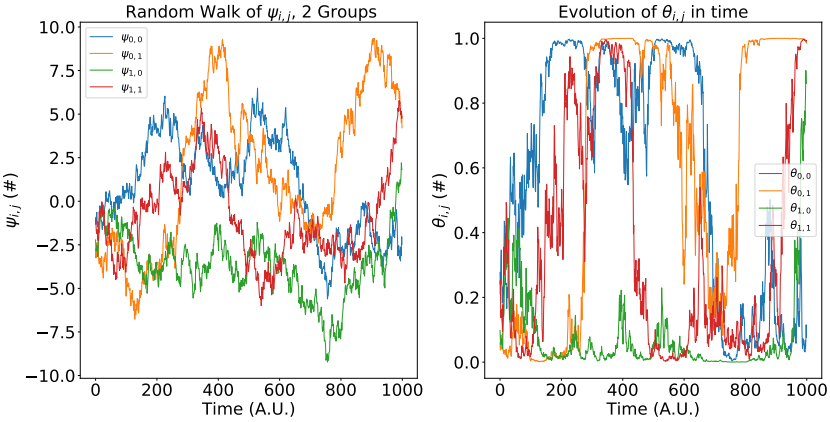

Figure 4.4: Evolution in time of ψ and θ, $s_{diag} = 0.1$ and $s_{snb} = 0.025$

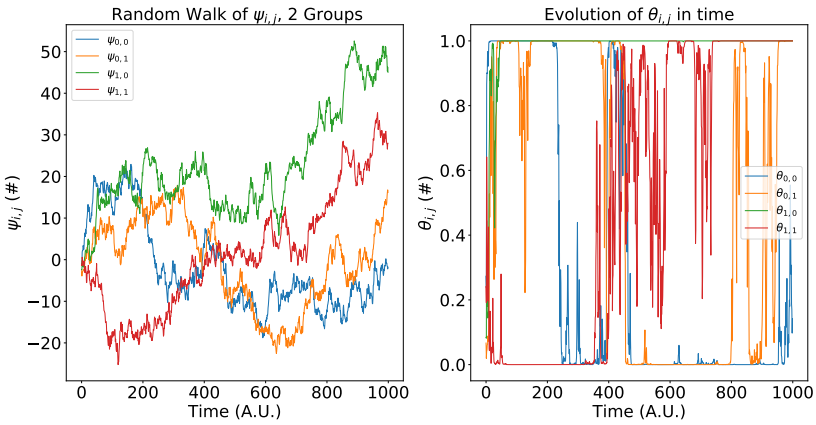

Figure 4.5: Evolution in time of ψ and θ, $s_{diag} = 1$ and $s_{snb} = 0.25$

when ψ^t crosses the range $[-4, 4]$, the coefficients of the affinity matrix are locked to the "absorbing states", 0 or 1. Concluding, this model is easy to simulate and useful for short time simulations but, is not a good model for long time simulations. This is also true when the entries of Γ^t have large values. In general, this phenomenon depends on the variance parameter which captures the diffusion behavior of the underlying process, thus determining the probability of being in one 'absorbing state' after given time steps.

4.3.2 Groups fixed and a varying number of nodes

In the simulation study presented in Xu and Hero [40], the authors focus on the case of affinity matrix of order $O(1)$, i.e. dense networks. As a consequence, the goodness of fit is really robust, in particular, the performance of the proposed inference method does not crucially depend on the estimation of Γ^t in the limit of dense networks. In the Xu and Hero model, the robustness of the estimation method to different values of Γ^t is related to the Extended Kalman Filter.

The robustness of the estimation and the EKF

In this subsubsection, we study the robustness of inference process proposed by Xu and Hero: in fact, the parameter estimation (θ and the labeling of the nodes) is weakly sensitive to a wrong estimate of Γ^t. To prove this statement, we study in detail the EKF algorithm and the term K^t. The equation of Kalman gain K^t in the Xu and Hero model is:

$$K^t = R^{t|t-1}(H^t)^T[H^t R^{t|t-1}(H^t)^T + \Sigma^t]^{-1} \tag{4.21}$$

where H^t is a diagonal matrix with entries equal to the derivative of the logistic function in Eq. 4.19, Σ^t is a diagonal matrix with entries equal to the variance of the observation noise z^t. $R^{t|t-1}$ is:

$$R^{t|t-1} = F^t R^{t-1|t-1}(F^t)^T + \Gamma^t \tag{4.22}$$

where $F^t = I$ is the state transition matrix, while the entries of Σ^t are:

$$(\sigma^2)^t_{ab} = \frac{h\big(\psi^{t|t-1}_{ab}\big)\big(1 - h\big(\psi^{t|t-1}_{ab}\big)\big)}{n_{ab}} \tag{4.23}$$

where n_{ab} is the number of possible links between the group a and group b.
If we consider a diagonal Γ^t for simplicity, all the matrices are diagonal and we can explicit the value of each component. Therefore the Eq. 4.21 becomes, for each diagonal entry:

$$\begin{aligned} k^t_{diag} &= \frac{(r + s_{diag})\frac{e^{-\psi}}{(1+e^{-\psi})^2}}{\big(\frac{e^{-\psi}}{(1+e^{-\psi})^2}\big)^2(r + s_{diag}) + \frac{1}{n}\frac{1}{1+e^{-\psi}}\big(1 - \frac{1}{1+e^{-\psi}}\big)} \\ &= \frac{(r + s_{diag})\frac{e^{-\psi}}{(1+e^{-\psi})^2}}{\big(\frac{e^{-\psi}}{(1+e^{-\psi})^2}\big)^2(r + s_{diag}) + \frac{1}{n}\frac{e^{-\psi}}{(1+e^{-\psi})^2}} \\ &= \frac{(r + s_{diag})}{\frac{e^{-\psi}}{(1+e^{-\psi})^2}(r + s_{diag}) + \frac{1}{n}} \\ &= \frac{n(r + s_{diag})(1 + e^{-\psi})^2}{ne^{-\psi}(r + s_{diag}) + (1 + e^{-\psi})^2} \\ &= \frac{(1 + e^{-\psi})^2}{e^{-\psi} + \frac{(1+e^{-\psi})^2}{n(r+s_{diag})}} \end{aligned} \tag{4.24}$$

where r and s_{diag} are the diagonal components of $R^{t-1|t-1}$ and Γ^t, while ψ is $\psi^{t|t-1}$. In the networks simulated by Xu and Hero, the number of nodes for each group is 32, so $n = 32*31 = 992$, $r \simeq 0.004$ and $s_{diag} = 0.01$. Since Xu and Hero impose $s_{diag} = 0.01$, we will refer to this value as a reference value for our analysis.

Using the result obtain in Eq. 4.24, we can study the dependence of k^t on s_{diag}, by varying the value of ψ^t. This analysis is summarized in Fig. 4.6 and 4.7: the first one shows the value of k^t for 5 different values of s_{diag} for $-4 \leq \psi \leq 4$, while the second one shows the ratio:

$$Ratio = \frac{k^t_{other}}{k^t_{0.01}} \tag{4.25}$$

where $k^t_{0.01}$ refers to the value of k^t calculate for $s_{diag} = 0.01$. In Fig. 4.6 and 4.7, we observe the the relation between k^t and ψ and we show the *Ratio* for a network with 32 nodes for each group, which is the same value used by Xu and Hero [40].

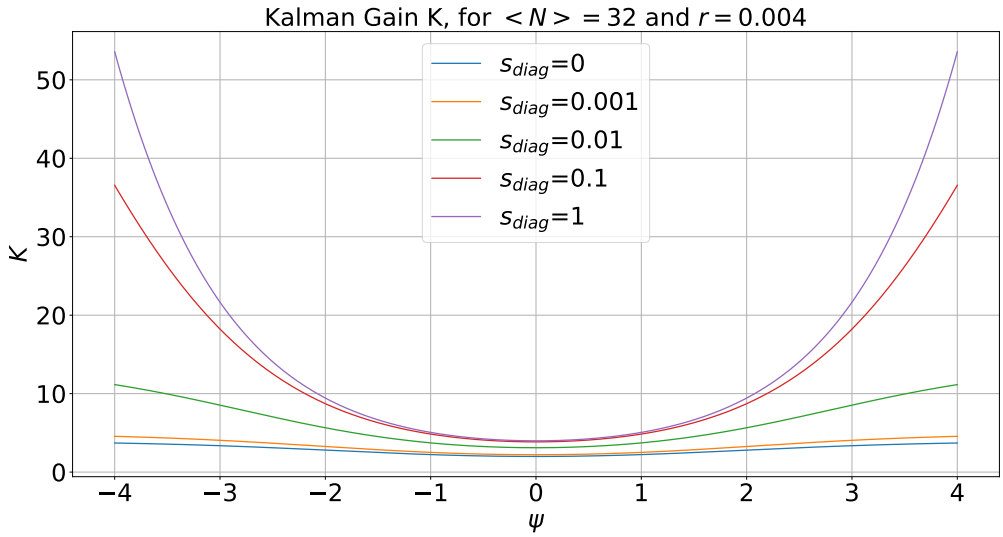

Figure 4.6: Kalman gain for $\langle N \rangle = 32$, $r = 0.004$

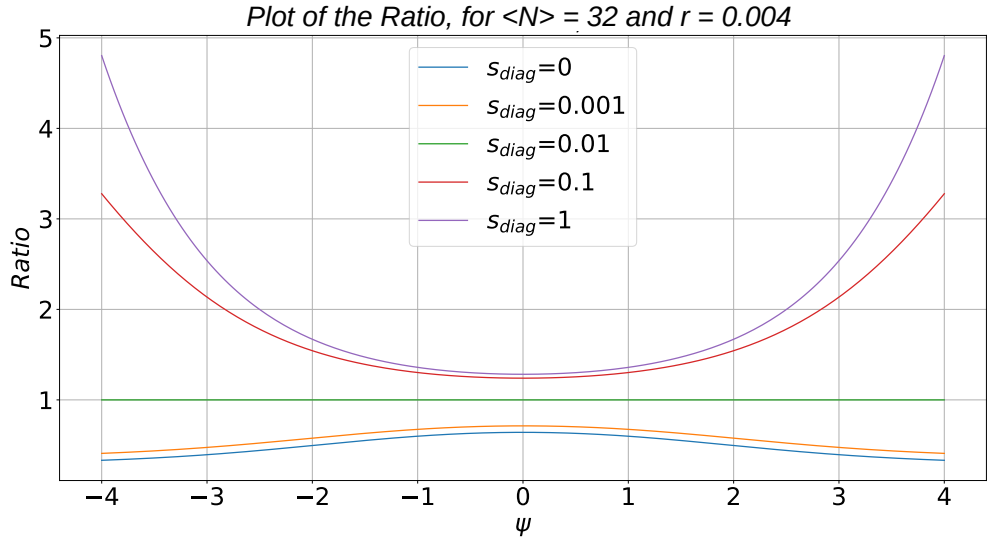

Figure 4.7: Ratio between Kalman gains for different s_{diag}, $\langle N \rangle = 32$, $r = 0.004$

As shown in Fig. 4.7, the ratio between the reference value of $s_{diag} = 0.01$ and the other values is included in range $[0.3 - 5]$ if $\psi = \pm 4$ but, for $-2 \leq \psi \leq 2$, the range is smaller ($[0.5 - 2]$). This proves why it is so difficult to estimate the entries of Γ^t, since a large variation in Γ^t does not affect strongly the estimate of k^t. In addition, a small error in the estimate of k^t is followed by a small error in the estimate of ψ^t, which is used to calculate the estimate of θ. Thus, a wrong initial estimate of Γ^t does not affect strongly the estimate of the parameter of the model, as also Xu and Hero show in their paper [40].

We can deepen this analysis,by generating a network with more nodes and imposing that the mean for each group is $\langle N \rangle = 100$: we observe that the distance between the ratios is reduced significantly, as we can observe in Fig. 4.8, 4.9:

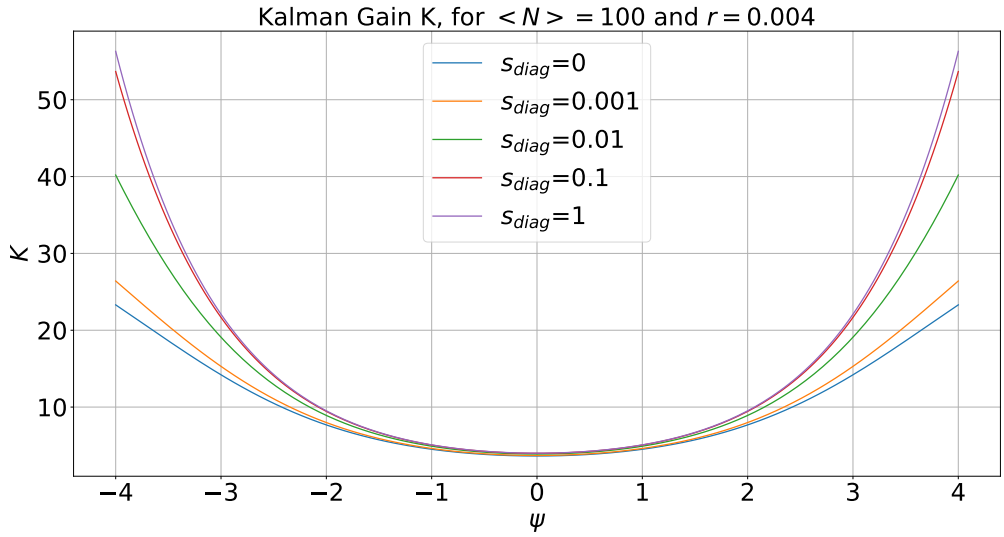

Figure 4.8: Kalman gain for $\langle N \rangle = 100$, $r = 0.004$

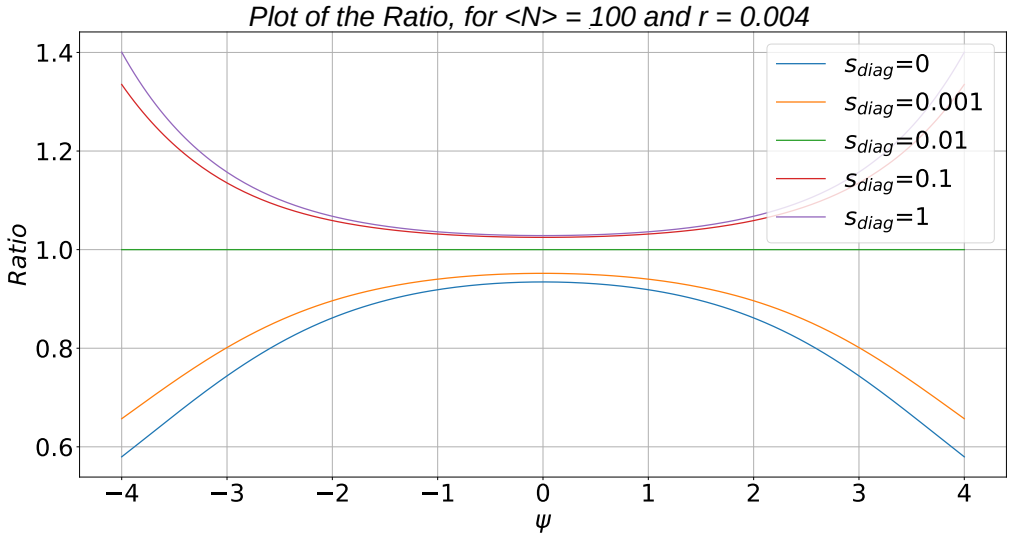

Figure 4.9: Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 100$, $r = 0.004$

Hence for any value of s_{diag}, the value of Kalman gain is similar for a large part of the range of ψ.
Thus, the inference of ψ dipends little on s_{diag} if the number of nodes in the network is large enough. Xu and Hero use a non diagonal matrix for Γ^t but, as we explain in Appendix A, the

results are not affected by this choice.

On the contrary, considering the case of sparse networks, the results suggest that the inference process is strongly dependent on the variation of s_{diag}, see Fig. 4.10, 4.11, 4.12, and 4.13.

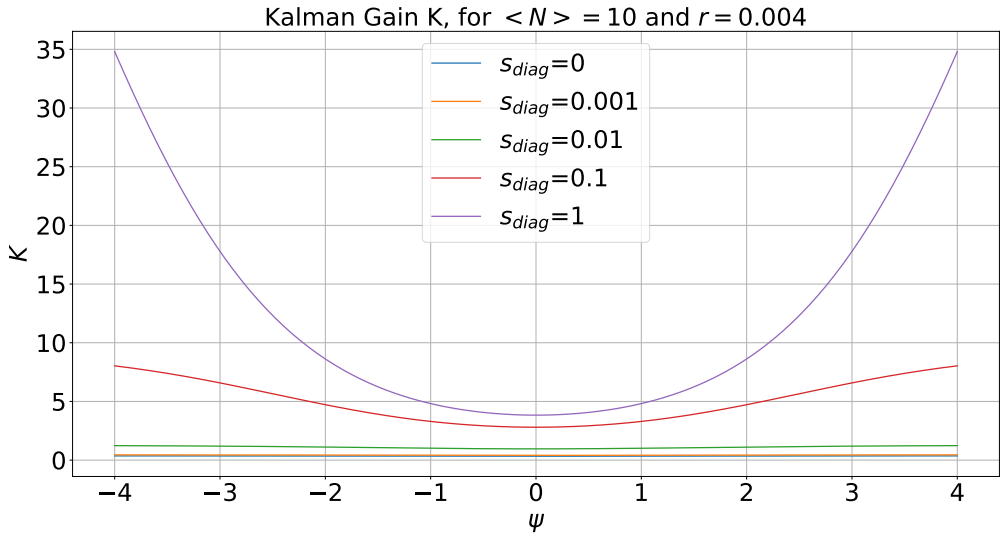

Figure 4.10: Kalman gain for $\langle N \rangle = 10$, $r = 0.004$

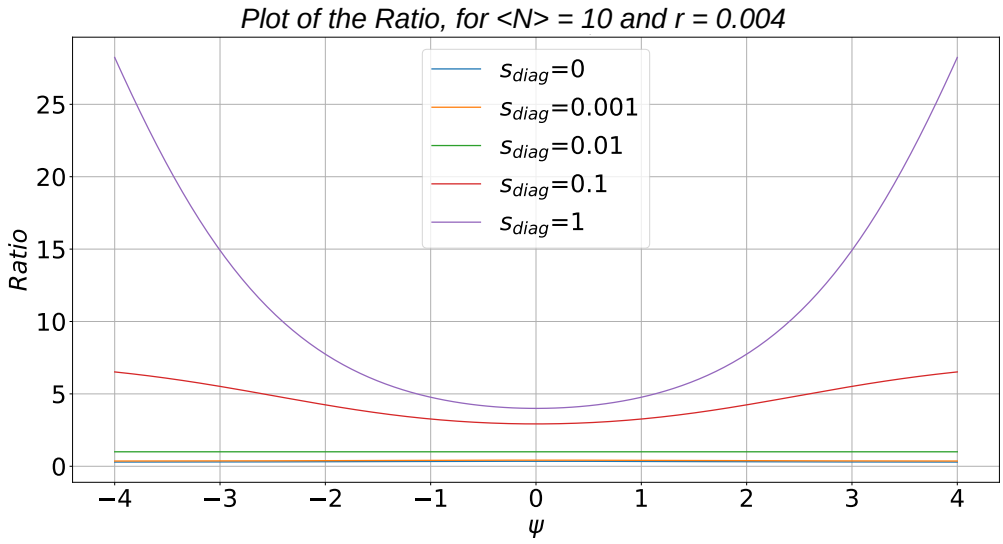

Figure 4.11: Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 10$, $r = 0.004$

For $\langle N \rangle = 10$, the values of K^t start to be very different and the ratio grows faster for multiples of s_{diag}.

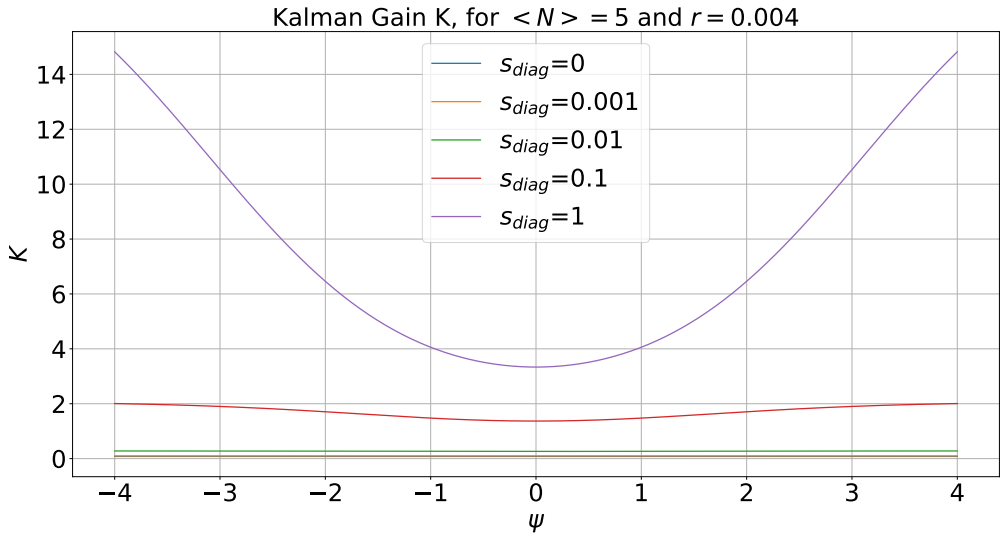

Figure 4.12: Kalman gain for $\langle N \rangle = 5$, $r = 0.004$

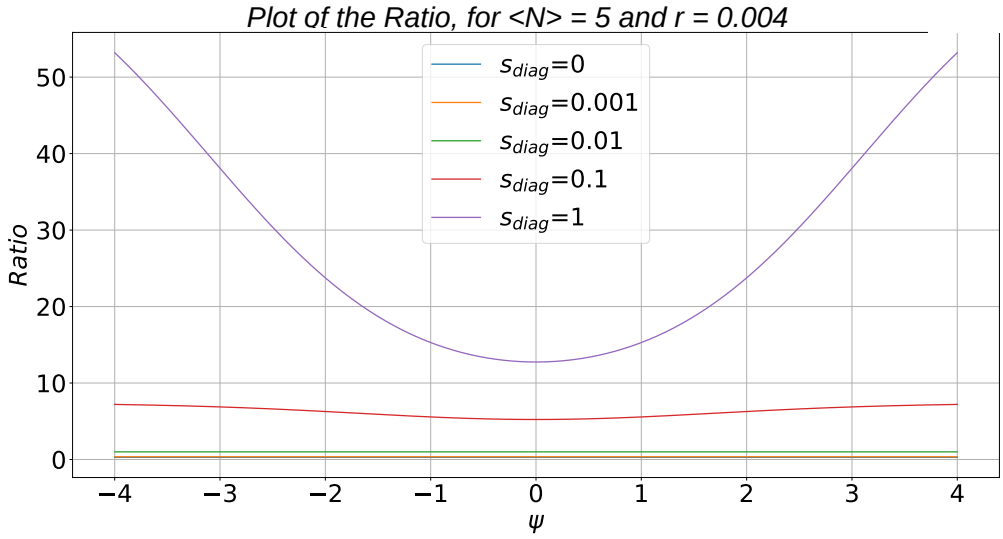

Figure 4.13: Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 5$, $r = 0.004$

For $\langle N \rangle = 5$ the value of s_{diag} is important to obtain the correct prediction since all the gains K^t are different.

In this subsection we focus on the relevance of hyperparameter Γ^t and why it is difficult to have a good estimate of it. In the following we study the effect on the metrics (MSE, ARI, and MSPE) for different Γ^t while varying the number of nodes for each group.

Mean Square Error

The mean square error is one of the metrics we used to test the model and it measures the mean squared difference between the estimated affinity matrix $\tilde{\Theta}^t$ and the affinity matrix adoped to generate the network Θ^t. We simulate a temporal network with 10-time steps and we calculate the mean of the MSE obtained at each step. This is the value of the MSE associated to that simulation.
We consider 50 simulations, then we reduce the number of nodes in the network by 10 until we reach the value of 20 nodes in the network: the goal is to analyze if the MSE depends on the number of nodes in the network while keeping the number of groups in the network fixed. We repeat the simulations three times, using three different values of Γ^t in the estimation process: the original Γ^t used to generate the network that has to be inferred, which is the same as used by Xu and Hero defined in Eq.4.17, and two other choices Γ^t, respectively $10\Gamma^t_{Xu}$ and $\frac{1}{10}\Gamma^t_{Xu}$. The first set of simulations is obtained with a network with 130 nodes and 2 groups. With 50 simulations we can obtain a good estimate of the mean and variance of the metric we are testing.

For a number of nodes $N > 60$ (Fig. 4.14), the value of MSE is of the same order of magnitude for all the number of nodes we use to generate the networks, even if it is slowly decreasing, as we can observe in Fig. 4.14. Furthermore, the value of MSE obtained is quite similar for the three different Γ^t. This result confirms the previous analysis on the Kalman gain K^t for a large number of nodes.

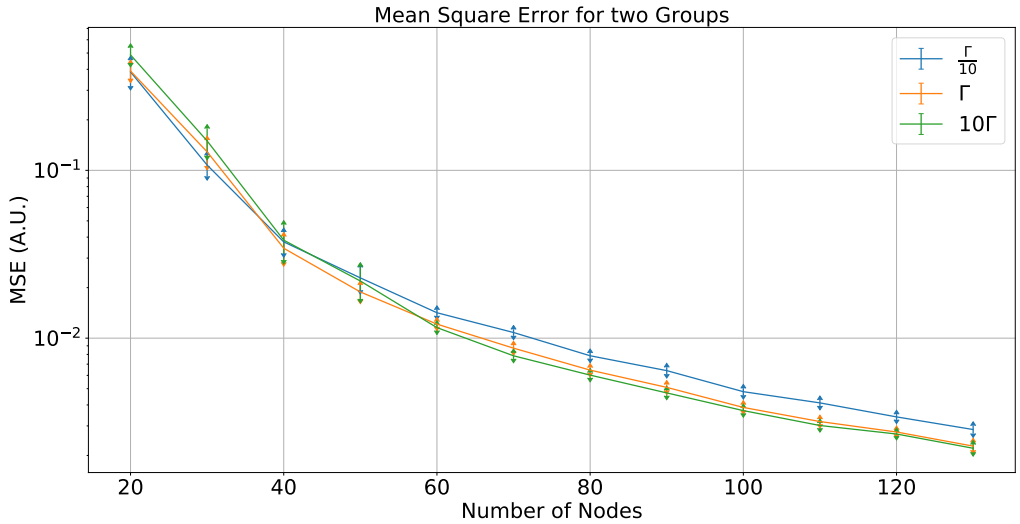

Figure 4.14: MSE for 2 groups and different Γ^t values.

For a number of nodes $40 < N < 60$, we observe a change for the three Γ^t. The correct value of Γ^t (the orange one) produces the best performance. For $N < 40$ we observe that:

- the MSE's grows faster if compared to networks with a larger number of nodes;
- The MSE for the three Γ^t is quite similar due to the small difference of the Kalman gain K^t for that density of nodes for each group 4.15;

This means that the use of the correct value of Γ does not compensate the estimation error due to a network which contains a small number of nodes.

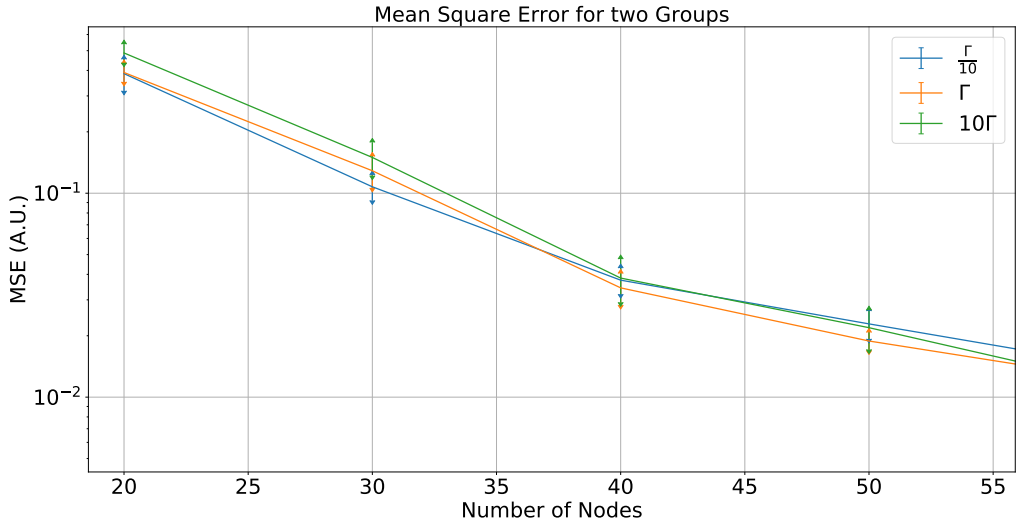

Figure 4.15: MSE for 2 groups and different Γ^t values, focus on the range $20 < N < 50$

Adjusted Rand Index

The analysis of the ARI is quite similar to the previous one. For all the number of nodes, its value is very similar for the three Γ^t (see Fig. 4.16) and only for $N = 20$ we can observe some discrepancy between different conditions, as shown in Fig. 4.17.

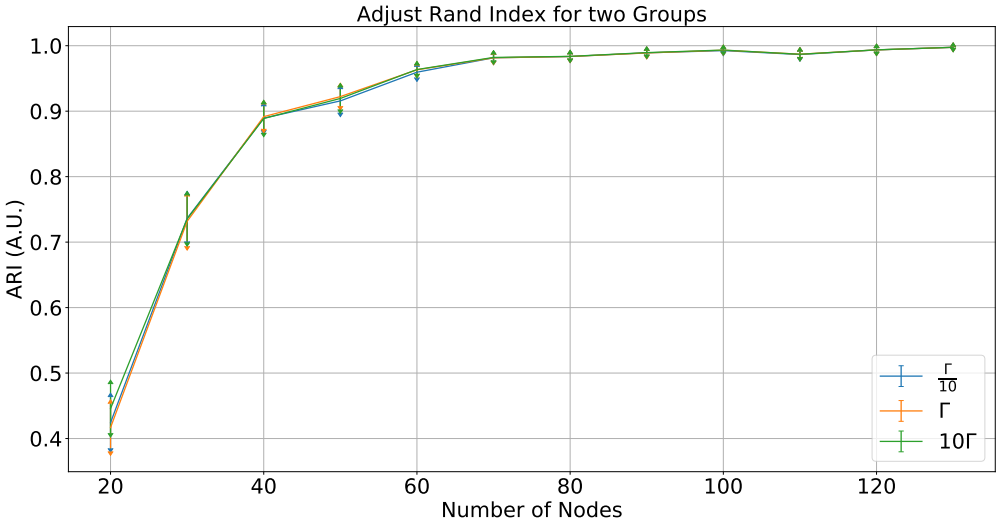

Figure 4.16: ARI for 2 groups and different Γ^t values

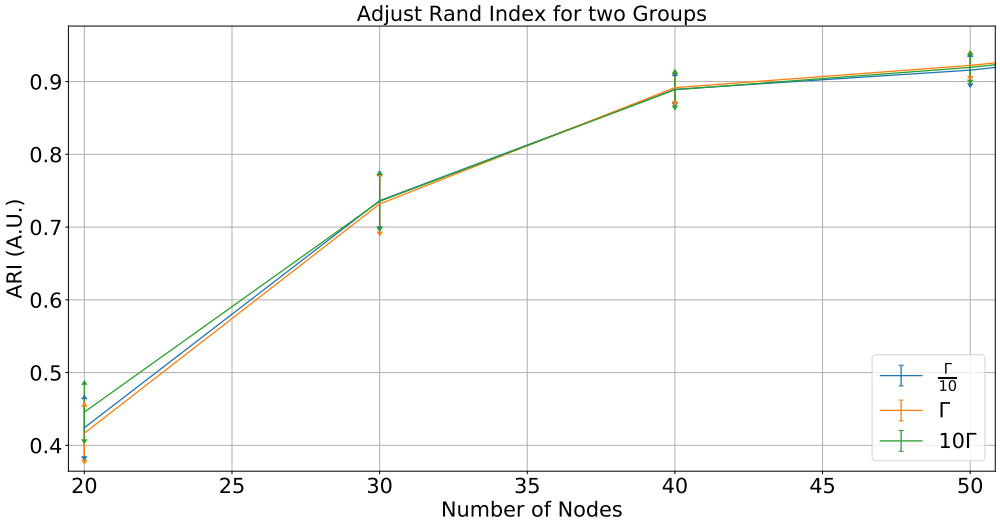

Figure 4.17: ARI for 2 groups and different Γ^t values, focus on the range $20 < N < 50$

Mean Square Predictor Error

We can observe in Fig. 4.18 that the estimation method gives the best results for submultiple of Γ^t, because of a lower value of the MSPE. The correct matrix Γ^t obtain the best score respect to the other Γ^t only for $N = 20$, as expected from the study of the Kalman gain.

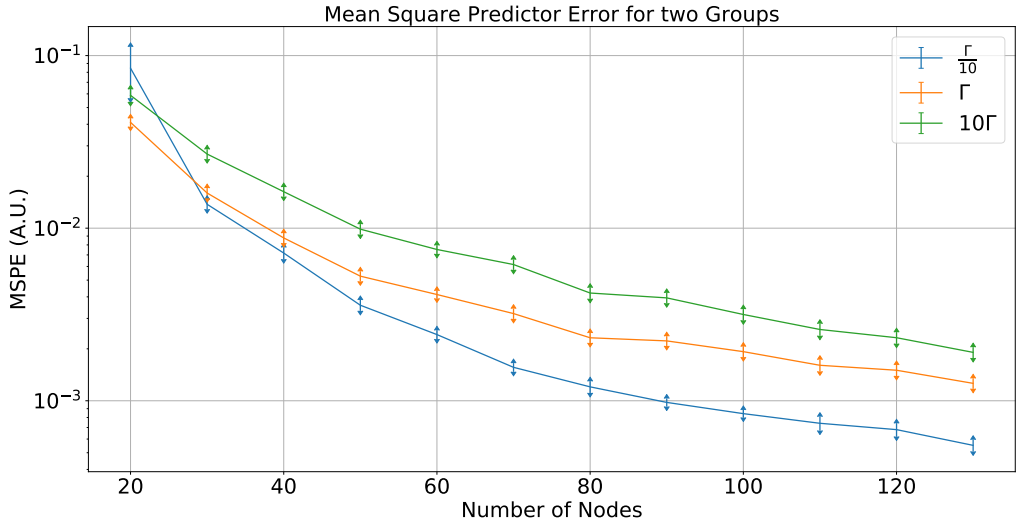

Figure 4.18: MSPE for 2 groups and different Γ^t values

4.3.3 Simulation study with varying number of groups

In this subsection, we analyze the case of keeping constant the number of nodes and progressively increasing the number of groups (both the cases have the same aim, that is reducing the density of nodes for each group).
This simulation study is computationally hard because of the performances of the label switching algorithm for increasing number of groups. Every time the number of groups is raised by one, the algorithm has to calculate the log posterior probability for a larger number of groups. This means that, for each node, we have a set of instructions that is repeated once more. Furthermore, this holds for all the time steps. For instance, if for two groups are required 30 seconds for each simulation, for 6 groups the time raises to 10 minutes. For this reason, we fix the value of six as the maximum number of groups in this simulation study.

Mean Square Error and Adjusted Rand Index

We can observe in Fig. 4.19 and 4.20 that, for both MSE and ARI, the correct value of Γ^t (the green one) has a better score if compared to the other values. If we compare the values of MSE and obtained for the different matrix Γ^t:

- For ARI, the scores obtained are very similar and almost independent from the matrix Γ^t used in the simulations;

- For MSE, the values obtained in the different simulation depends on the matrix Γ^t used to infer the network.

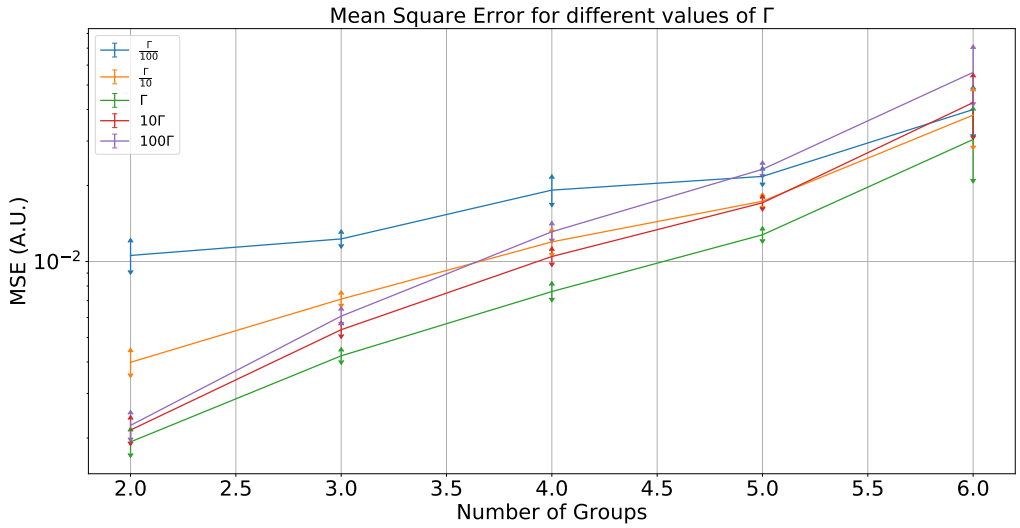

Figure 4.19: MSE for 128 nodes and different Γ^t values

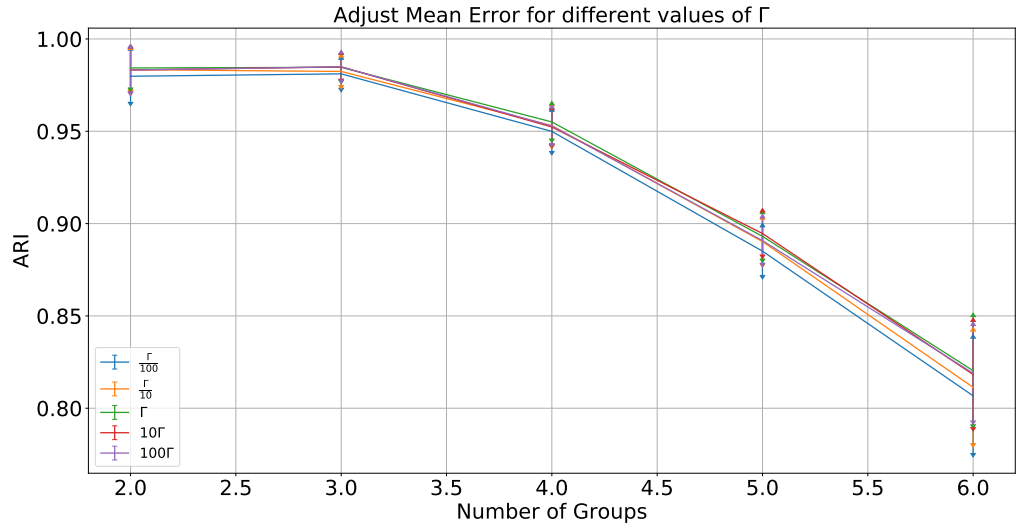

Figure 4.20: ARI for 128 nodes and different Γ^t values

Mean Square Predictor Error

In this case we note that the correct matrix Γ^t does not obtains the best scores, unlike what we observed before. This fact could be explained studying the number of nodes for each group. In

fact, when the network contains six groups, everyone of them is composed by 21 nodes. In Fig. 4.18, we can observe that the Mean Square Predictor Error of the correct value of Γ^t obtains the best score only for a density equal to 10 nodes for each group. To obtain the same density, we should consider 13 groups, but this analysis would require a very long computational time. Hence, we decide to limit the study with 6 groups, which was the reachable limit for our instruments.

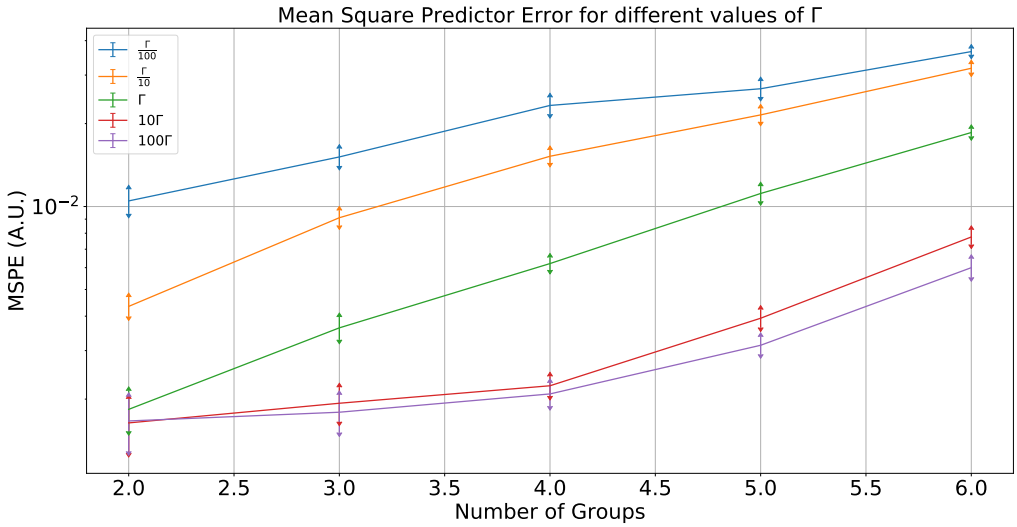

Figure 4.21: MSPE for 128 nodes and different Γ^t values

4.3.4 Memory effect

The algorithm for the label inference uses the previously estimated assignments to predict the next ones. We know that at each time step a certain percentage of nodes leaves their community to be reallocated in another one, so the label switching algorithm is based on the fact that the majority of the population of the nodes remains in their class between time steps. This is a *memory effect*. When the percentage of leaving nodes equal to 10%, the label switching algorithm has good performances as in Xu and Hero. We want to investigate what happens if we raise this percentage. The analysis uses a network with 3 groups and 33 nodes for each group.

ARI

We decide to analyze the ARI for a percentage that varies in the range $[0.02 - 0.29]$. When the percentage is higher, e.g. 50%, some problems of group identifiability may occur, independently from the adopted algorithm. We use the spectral clustering as a benchmark since this algorithm does not depend on the communities of the previous time step, thus its performance does not

change if we vary the percentage. For each network generated, we use both Xu and Hero model and SSBM spectral clustering to infer the parameter of the network. We generate 30 networks for each percentage of nodes leaving their group between time steps, to obtain a good estimate of the mean ARI and its related variance. In Fig. 4.22, we can observe the difference between SSBM spectral clustering and Xu and Hero algorithm: the Pearson coefficient ρ_{XY} for SSBM is clearly null as expected, while for Xu and Hero model the coefficient shows a weakly negative linear correlation.
So the *memory effect* is slowly decreasing for a raising percentage of nodes that leave their class. This means that the higher the percentage, the more the algorithm's result is similar to the static observation.

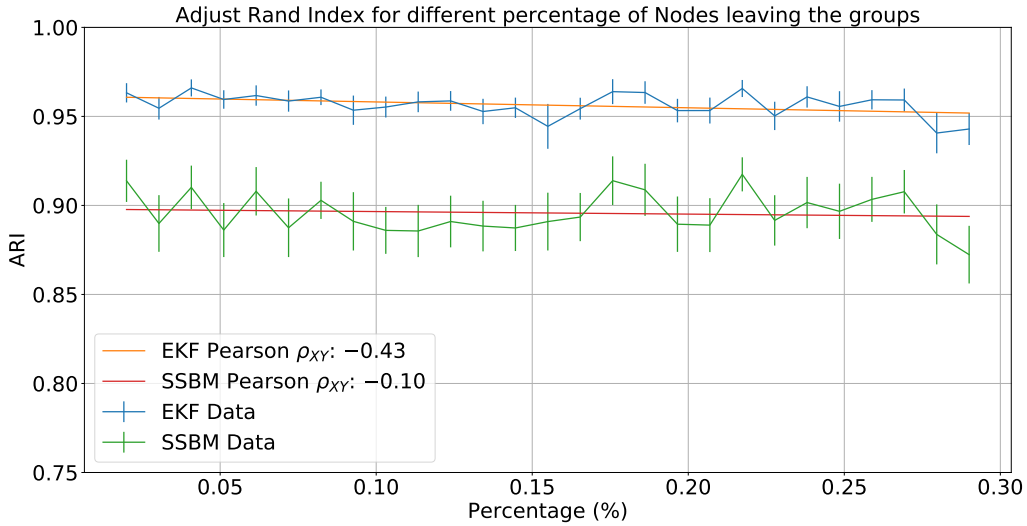

Figure 4.22: ARI evolution varying the percentage of leaving nodes

MSE

Investigating MSE, the results of the comparison in Fig. 4.23 are similar to the previous one: SSBM results are not correlated, while Xu and Hero's results are weakly correlated. This means that the difference between the predicted ψ^t and the real one grows slowly.

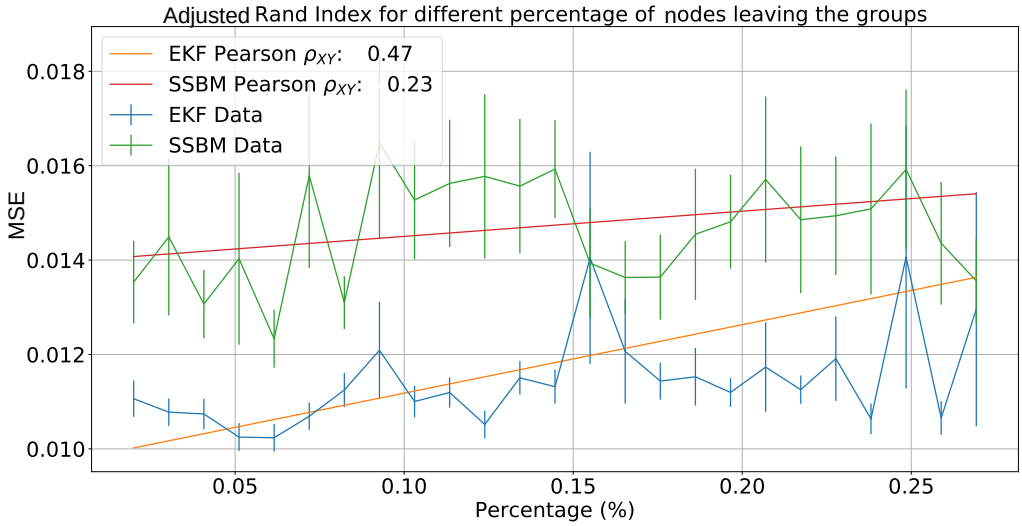

Figure 4.23: MSE evolution varying the percentage of leaving nodes

4.4 Conclusion

In this chapter, we have pointed out some drawbacks of the Xu and Hero model. We group them into two different categories:

- Analytical model problems: the Xu and Hero generative model, shown in Eq. 4.1, exhibits the absorbing state phenomenon, as we demonstrate in section 4.3.1. This fact exclude the use of this model for long time simulations. Furthermore, the authors impose several constraints to the hyperparameters ($F^t = I$, fixed structure for the matrix Γ^t).

- Simulations problems: the results depend little on the matrix Γ^t employed in the inference process. This fact implies a low possibility to infer the correct value of Γ^t. Furhtermore, we observe in section 4.3.4 a memory effect of the algorithm used in the inference process.

In the following chapter, we propose two generalizations of the model to overcome these criticalities, with a particular focus to applications. We combine Xu and Hero model with the two-time series models we analyzed in Chapter 2, the AR(1) model and the VAR(1) model:

- Both AR(1) and VAR(1) models are mean-reverting, thus the network dynamics does not display 'absorbing states' for the link probabilities;

- We can employ the properties of these two models to collect more informations of the graph while inferring its parameters: in Xu and Hero model, we have to impose some

constraints to the structure of Γ^t and the matrix F^t must be known a priori; viceversa, for the generalizations, we do not need to impose these constraints.

Chapter 5

Generalization of Xu and Hero Model to Auto-Regressive Processes

In this chapter, we propose two novel dynamic stochastic block models which overcome the issues highlighted for the model of Xu and Hero [40]. In particular, in the first section and the second section we exploit the framework introduced by Xu and Hero, but describing the evolution of the affinity matrix with an autoregressive process, either AR(1) or VAR(1). We also introduce a novel inference algorithm for model estimation. In the third section and in the fourth section, we study the results obtained in the numerical simulations, similarly to the analysis presented in the previous chapter. The simulation study presented in this chapter is to assess the performances of the estimation methods of the novel algorithms. In the last section, we compare the results obtained for the two new models. All the results presented in this chapter are original.

5.1 AR-SBM Model

We have analyzed Auto-Regressive processes in Chapter 2. Our purpose is to generalize the equations describing the evolution of the affinity matrix in Xu and Hero model with an AR(1) process.
Xu and Hero propose this system of equations for the evolution of $\boldsymbol{\psi}$, i.e. the logit of the vectorized affinity matrix $\boldsymbol{\theta}$:

$$\begin{cases} \boldsymbol{\psi}^t = F^t \boldsymbol{\psi}^{t-1} + \boldsymbol{v}^t \\ \boldsymbol{Y}^t = h(\boldsymbol{\psi}^t) + \boldsymbol{z}^t \end{cases} \tag{5.1}$$

where F^t is the identity matrix, $\boldsymbol{v}^t$ is the noise process, $h(x)$ is the logistic function and $\boldsymbol{z}^t$ the observation noise. Focusing only on the process equation, each component evolves like:

$$\psi_{ab}^t = \psi_{ab}^{t-1} + v_{ab}^t \tag{5.2}$$

Hence, Xu and Hero model assumes a random walk for ψ_{ab}, which is not a stationary process, thus the probability of each component of $\boldsymbol{\psi}$ becoming larger and larger as time goes on is very high. The results of this are that θ, the affinity matrix, becomes a matrix of values really close to 1 and 0. In real world networks, it is very unlikely to observe this behavior.
As a generalization, we consider the affinities evolving as an AR(1) model:

$$\psi^t = a + b\psi^{t-1} + v^t \tag{5.3}$$

The process is stationary when $|b| < 1$. We assume that the matrix B and the array $\boldsymbol{A}$ are time-independent:

$$B^t = B \quad \boldsymbol{A}^t = \boldsymbol{A}$$

The covariance matrix Γ is also time-independent as in the Xu and Hero model. The process mean μ is:

$$\mu(\psi^t) = \frac{a}{1-b} \tag{5.4}$$

When we consider each ψ^t_{ab} evolving independently from each other ones, then the evolution of $\boldsymbol{\psi}^t$ is described by:

$$\boldsymbol{\psi}^t = \boldsymbol{A} + B\boldsymbol{\psi}^{t-1} + \boldsymbol{v}^t \tag{5.5}$$

With

$$B = \begin{pmatrix} b_{1,1} & 0 & \cdots & 0 \\ 0 & b_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_{n,n} \end{pmatrix}$$

and Γ, i.e. the covariance matrix of the process noise v_t:

$$\Gamma = \begin{pmatrix} \Gamma_{1,1} & 0 & \cdots & 0 \\ 0 & \Gamma_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Gamma_{n,n} \end{pmatrix}$$

Here, differently, from Xu and Hero, the noise components are uncorrelated.

5.1.1 Model's hyperparameters

The hyperparameters in this generalized model are:

1. The mean $\boldsymbol{\mu}^1$ of the initial state $\boldsymbol{\psi}^1$;
2. The covariance matrix Γ^1 of the initial state $\boldsymbol{\psi}^1$;

3. The covariance matrix Σ^t of the observation noise $\boldsymbol{z}^t$;
4. The covariance matrix Γ^t of the process noise $\boldsymbol{v}^t$;
5. The state evolution matrix B;
6. The vector of parameters $\boldsymbol{A}$.

Indeed, the model is more flexible if compared with the Xu and Hero model. To infer these two new hyperparameters, we use an Expectation-Maximization algorithm:

1. Using SSBM spectral clustering initialization we obtain initial estimate $\boldsymbol{c}^t$, the class memberships at time t for all time steps;
2. Compute all block densities Y^t using W^t and $\boldsymbol{c}^t$;
3. Compute all ψ^t using block densities Y^t;
4. Estimate the AR(1) process, thus obtaining an estimate of $\bar{a}_i$, $\bar{b}_{i,i}$ and $\bar{\Gamma}_{i,i}$, by conditioning on $\Psi_i = \{\psi_i^0, \psi_i^1, \cdots, \psi_i^t\}$;
5. while (true):
6. Compute all $\hat{\psi}$;
7. Estimate the AR(1) process, thus obtaining an estimate of $\hat{a}_i$, $\hat{b}_{i,i}$ and $\hat{\Gamma}_{i,i}$ by conditioning on $\hat{\Psi}_i = \{\hat{\psi}_i^1, \hat{\psi}_i^2, \cdots, \hat{\psi}_i^t\}$;
8. if $||\hat{A} - \bar{A}||_2 < threshold$ and $||\hat{b_{i,i}} - \bar{b_{i,i}}||_2 < threshold \rightarrow$ break;
9. else: $[\bar{a}_i, \bar{b}_{i,i}] \leftarrow [\hat{a}_i, \hat{b}_{i,i}]$.

While repeating the instruction from 6 to 9, we can observe that the fit of $\hat{a}_i$, $\hat{b}_{i,i}$ and $\hat{\Gamma}_{i,i}$ is made removing the initial estimate ψ_i^0. This is done because ψ_i^0 is the result of spectral clustering algorithm also for Xu and Hero model used in instruction 6.
Therefore, the initial point of the estimation process is fixed at the initial step of the inference process, thus it is not considered in the estimation of $\hat{a}_i$ and $\hat{b}_{i,i}$ in order to reduce the estimation error.

The estimate of $\hat{a}_i$, $\hat{b}_{i,i}$ and $\hat{\Gamma}_{i,i}$ are obtained by solving the Eq. 2.8 and Eq. 2.11.

5.2 VAR-SBM model

The VAR(1) model is a generalization of the AR(1) model. In AR(1) model, each component of $\boldsymbol{\psi}^t$ is independent from the others. Thus, if the network is divided into k groups, the model dynamics is described by a set k^2 independent equations.
Within this model, we aim to describe the dependence structure between the components of $\boldsymbol{\psi}^t$. Formally, the state equation is the same as before, i.e.

$$\boldsymbol{\psi}^t = \boldsymbol{A}^t + B^t \cdot \boldsymbol{\psi}^{t-1} + \boldsymbol{v}^t \tag{5.6}$$

where B^t is a generic matrix of dimension $k^2 \times k^2$, if k is the number of group of the system. The VAR process is not always stationary.
Furthermore, these matrices and $\boldsymbol{A}^t$ are time-independent, i.e.:

$$\boldsymbol{A}^t = \boldsymbol{A} \quad B^t = B \quad \Gamma^t = \Gamma \tag{5.7}$$

In this model, we impose that the matrix B^t is constructed using the same criteria proposed by Xu and Hero to construct the covariance matrix Γ^t, which correlates the components of the affinity matrix that share the same row or column. Thus, the matrices B^t and Γ^t share the same structure.

5.2.1 Model's hyperparameters

For this model, the hyperparameters are:

1. The mean $\boldsymbol{\mu}^1$ of the initial state $\boldsymbol{\psi}^1$;
2. The covariance matrix Γ^1 of the initial state $\boldsymbol{\psi}^1$;
3. The covariance matrix Σ^t of the observation noise $\boldsymbol{z}^t$;
4. The covariance matrix Γ^t of the process noise $\boldsymbol{v}^t$;
5. The static array $\boldsymbol{A}$;
6. The static matrix B.

The hyperparameters from 1 to 3 are inferred as described before. The estimate of $\boldsymbol{A}$, B and Γ are as follows.

Estimate of A and B

Using the properties that each $\boldsymbol{\psi}^t$ depends only on the previous state, we can write the equations:

$$E[\bar{\boldsymbol{\psi}}^t \bar{\boldsymbol{\psi}}^{t-1}] = B \cdot E[\bar{\boldsymbol{\psi}}^{t-1} \bar{\boldsymbol{\psi}}^{t-1}] + E[\boldsymbol{v}^t \bar{\boldsymbol{\psi}}^{t-1}] \tag{5.8}$$

where $\bar{\boldsymbol{\psi}}$ is $\bar{\boldsymbol{\psi}} = \boldsymbol{\psi} - \underline{\mu}(\boldsymbol{\psi})$.
Let us define $E[\bar{\boldsymbol{\psi}}^{\boldsymbol{t}} \boldsymbol{\psi}^{\boldsymbol{t-1}}] = \beta^1$ and $E[\boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}} \boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}}] = \beta^0$ and because of $\boldsymbol{v}^{\boldsymbol{t}}$ and $\boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}}$ are uncorrelated, it is $E[\boldsymbol{v}^{\boldsymbol{t}} \boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}}] = 0$. Thus, Eq. 5.8 becomes:

$$\beta^1 = B \cdot \beta^0 \tag{5.9}$$

And the matrix B is:

$$B = \beta^1 \cdot (\beta^0)^{-1} \tag{5.10}$$

Now, if we take the square of Eq. 5.8 and we take expectation:

$$E[\bar{\boldsymbol{\psi}}^{\boldsymbol{t}} \bar{\boldsymbol{\psi}}^{\boldsymbol{t^T}}] = E[(B \cdot \boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}} + \boldsymbol{v}^{\boldsymbol{t}})((B \cdot \boldsymbol{\psi}^{\bar{\boldsymbol{t-1}}} + \boldsymbol{v}^{\boldsymbol{t}})^T] \tag{5.11}$$

Using the previous notation and the properties of VAR model:

$$\beta^0 = B \cdot \beta^0 \cdot B^T + \Gamma_t \tag{5.12}$$

Where Γ_t is the covariance matrix $\boldsymbol{v}^t$. It is

$$\Gamma_t = \beta^0 - B \cdot \beta^0 \cdot B^T \tag{5.13}$$

To estimate the vector of parameters $\boldsymbol{A}$, we take the expectation of $\boldsymbol{\psi}$ as described by the state equation:

$$E(\boldsymbol{\psi}^t) = A + B \cdot E(\boldsymbol{\psi}^{t-1}) + E(\boldsymbol{v}^t) \tag{5.14}$$

By using the property of the stationary processes, i.e. $E(\boldsymbol{\psi}^t) = E(\boldsymbol{\psi}^{t-1}) = \boldsymbol{\mu}$, and knowing that $E(\boldsymbol{v}^t) = 0$ for Gaussian noise, the Eq. 5.14 becomes:

$$\boldsymbol{\mu} = \boldsymbol{A} + B \cdot \boldsymbol{\mu} \tag{5.15}$$

Thus, $\boldsymbol{A}$ is:

$$\boldsymbol{A} = \boldsymbol{\mu} - B \cdot \boldsymbol{\mu} \tag{5.16}$$

We can see that the estimate of $\boldsymbol{A}$, B and Γ^t depends only on β^0, β^1, and Γ. We can β^0, β^1 as:

$$\beta^0 = \begin{pmatrix} \frac{1}{T-1}\sum_{t=1}^{T} \psi_1^t \psi_1^t & \frac{1}{T-1}\sum_{t=1}^{T} \psi_1^t \psi_2^t & \cdots & \frac{1}{T-1}\sum_{t=1}^{T} \psi_1^t \psi_{k^2}^t \\ \frac{1}{T-1}\sum_{t=1}^{T} \psi_2^t \psi_1^t & \frac{1}{T-1}\sum_{t=1}^{T} \psi_2^t \psi_2^t & \cdots & \frac{1}{T-1}\sum_{t=1}^{T} \psi_2^t \psi_{k^2}^t \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{T-1}\sum_{t=1}^{T} \psi_{k^2}^t \psi_1^t & \frac{1}{T-1}\sum_{t=1}^{T} \psi_{k^2}^t \psi_2^t & \cdots & \frac{1}{T-1}\sum_{t=1}^{T} \psi_{k^2}^t \psi_{k^2}^t \end{pmatrix} \tag{5.17}$$

$$\beta^1 = \begin{pmatrix} \frac{1}{T-1}\sum_{t=1}^{T}\psi_1^t\psi_1^{t-1} & \frac{1}{T-1}\sum_{t=1}^{T}\psi_1^t\psi_2^{t-1} & \cdots & \frac{1}{T-1}\sum_{t=1}^{T}\psi_1^t\psi_{k^2}^{t-1} \\ \frac{1}{T-1}\sum_{t=1}^{T}\psi_2^t\psi_1^{t-1} & \frac{1}{T-1}\sum_{t=1}^{T}\psi_2^t\psi_2^{t-1} & \cdots & \frac{1}{T-1}\sum_{t=1}^{T}\psi_2^t\psi_{k^2}^{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{T-1}\sum_{t=1}^{T}\psi_{k^2}^t\psi_1^{t-1} & \frac{1}{T-1}\sum_{t=1}^{T}\psi_{k^2}^t\psi_2^{t-1} & \cdots & \frac{1}{T-1}\sum_{t=1}^{T}\psi_{k^2}^t\psi_{k^2}^{t-1} \end{pmatrix} \tag{5.18}$$

where ψ_i is the $i-th$ component of $\boldsymbol{\psi}^t$.

The algorithm used to infer all the hyperparameters is the same as before, by implementing the estimation of the 3 hyperparameters $\boldsymbol{A}$, B and Γ^t with the Eq. 5.10, 5.13, and 5.16.

5.3 AR-SBM Model Results

In this section, we analyze the numerical results of the AR-SBM model. We first employ this model as a generative model for temporal networks, then we analyze how the model could be used to infer the hyperparameters of a temporal network under different conditions. Let us notice that the estimation performance depends strongly on the number of time steps, differently from the model proposed by Xu and Hero.

5.3.1 Network creation

To compare the results obtained in Xu and Hero model with the results obtained in AR-SBM model, we set the initial affinity matrix equal to the one used in Xu and Hero. The main difference between the two models is the evolution of $\boldsymbol{\psi}^t$ in time: in Xu and Hero $\boldsymbol{\psi}^t$ performs a random walk. In AR-SBM model, the dynamic is autoregressive characterized by mean reversion. The smaller each entry of Γ^t, the closer the evolution of ψ^t will be to the process mean.

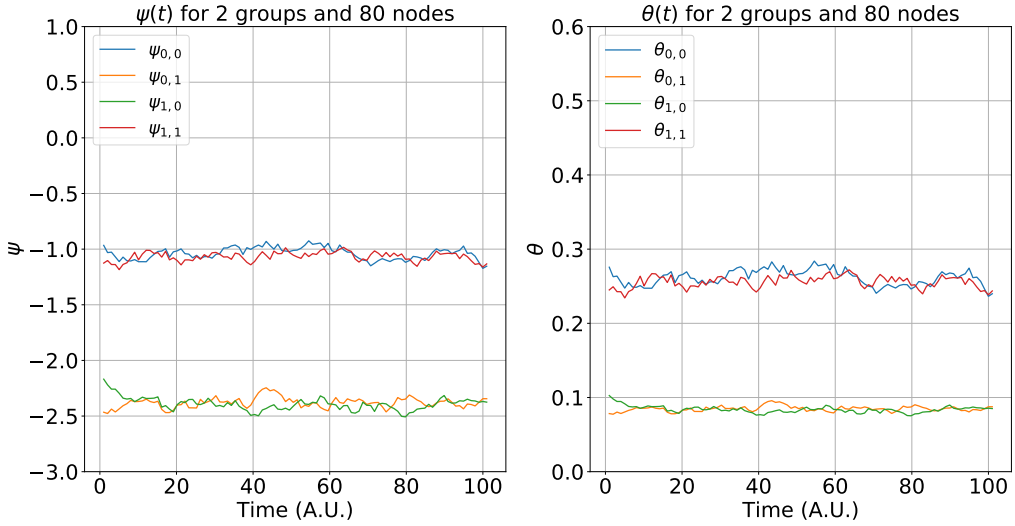

Figure 5.1: Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.001$

The hyperparameters we have to set are 3 for each component of ψ: A_i, $B_{i,i}$ and $\Gamma_{i,i} = s_{diag}$. Using Eq. 5.3, we decide to set the mean process of each entry of the affinity matrix equal to the same value employed by Xu and Hero. Since we can use many combinations of A_i and $B_{i,i}$ to obtain these values, we choose the following:

$$\begin{cases} A_i = -0.2112 \quad B_{i,i} = 0.8 \implies \mu(\psi_i^t) = -1 \\ A_i = -0.4780 \quad B_{i,i} = 0.8 \implies \mu(\psi_i^t) = -2.4 \end{cases} \tag{5.19}$$

The first equation is used for the diagonal entries of the affinity matrix, while the second the off-diagonal entries.

The value of s_{diag} is set equal to a set of powers of ten, from 10^4 to 10^0. As we observe in Fig. 5.1, for $s_{diag} = 0.001 = s_{diag}^{Xu}/10$, $\boldsymbol{\psi}^t$ fluctuates in a small range around the process mean.

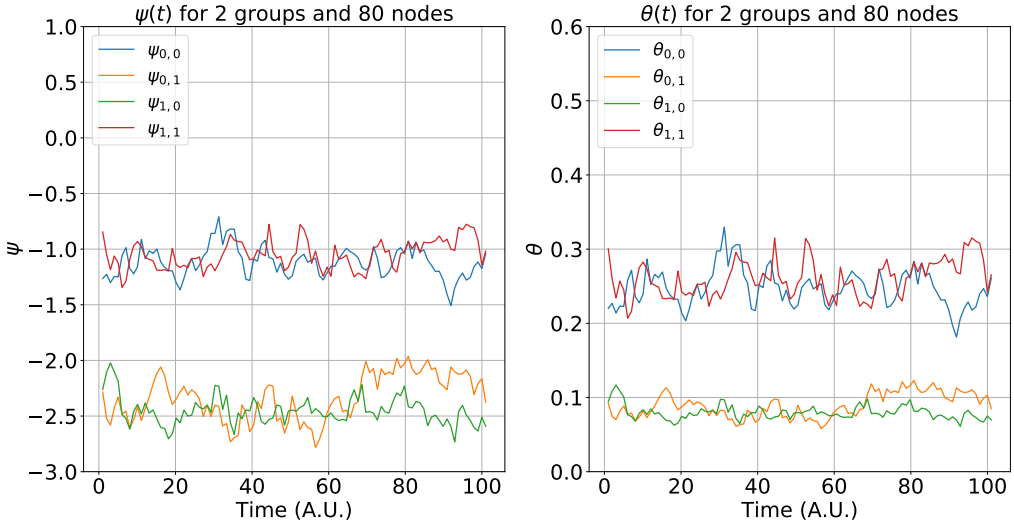

Figure 5.2: Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.01$

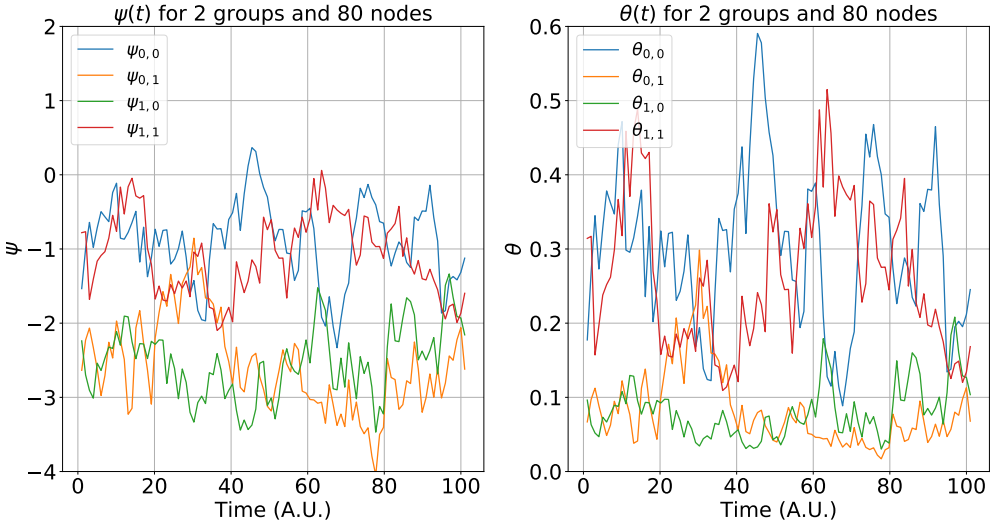

Figure 5.3: Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.1$

If we increase the value of s_{diag}, the fluctuations around the process mean are more prounced, as we can observe in Fig. 5.2 and Fig. 5.3.

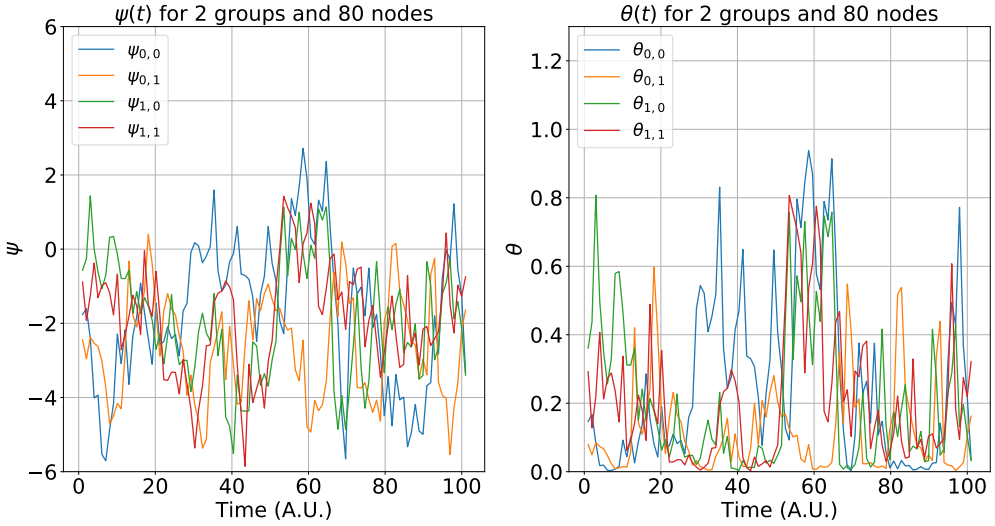

Figure 5.4: Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 1$

The most important property of this generative model is the stationarity, which allows to simulate an indefinitely long time dynamic network.

5.3.2 Inference of the model's hyperparameters

In this subsection we analyze the inference of the hyperparameters of the AR-SBM model while varying the number of time steps. The data are acquired for 5 different numbers of time steps, from 20 to 100, and for each number of time steps, we collect 30 simulations. Thus, for each couple composed by number of time steps and number of nodes, the 30 generated networks are very similar, since they are generated with the evolution model described in Eq. 5.2, which describes an AR process, and by using the same affinity matrix. Hence, the inferred hyperparameters are very similar between the different simulation: for this reason, both the estimation errors for $\boldsymbol{\psi}^t$ and for the value of the metrics (ARI and MSE) are small.

ARI

The first metric we analyze is the ARI. As we observe in Fig. 5.5, the difference of ARI between the simulation with 80 nodes and 120 nodes is very small. Also for 40 nodes, the score is very high and only for 20 nodes ARI decreases significantly.
If we compare the results by varying the number of time steps available to estimate the model, we observe a slightly decreasing trend of ARI for a lower number of time steps. The variation between 100-time steps and 20-time steps is small, suggesting that this metric depends more on the label switching algorithm than the EM algorithm used to infer $\boldsymbol{A}$ and B.

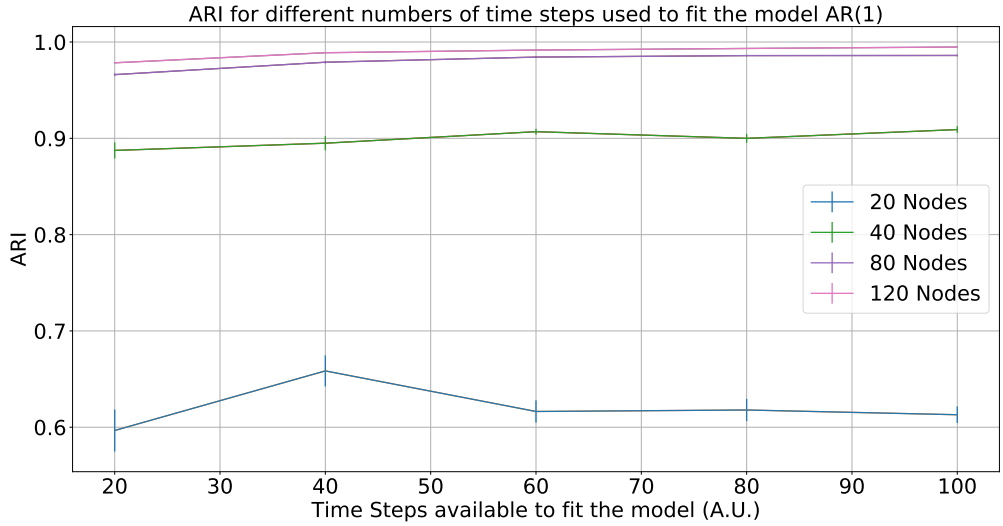

Figure 5.5: Evolution of ARI for different numbers of nodes

MSE of ψ^t

The analysis of the MSE of ψ^t is similar to the previous one for ARI: the metric depends more on the number of nodes rather than on the number time steps used to infer the hyperparameters of the simulation.

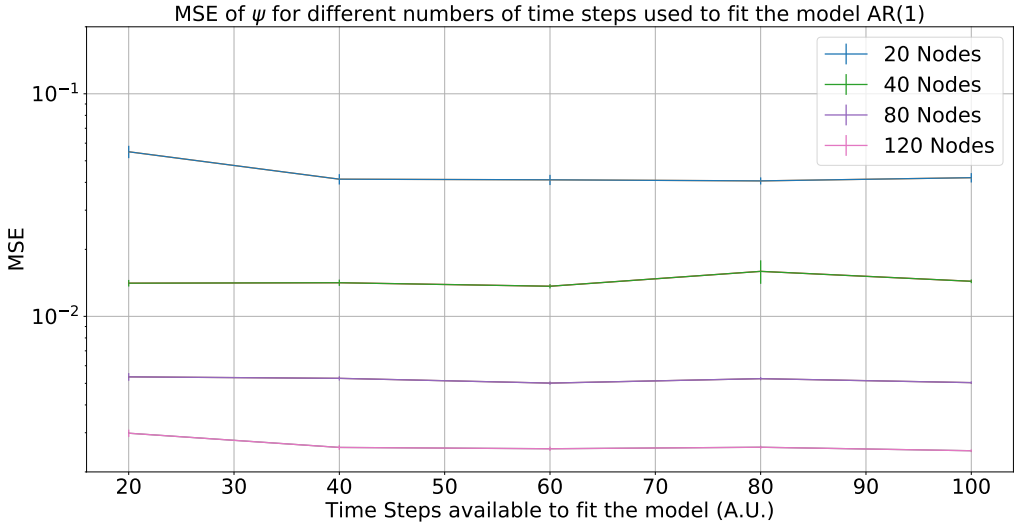

Figure 5.6: Evolution of MSE of ψ^t for different numbers of nodes

MSE of A, B, and Γ

As we observe in Fig. 5.7 and 5.8, the MSE associated with $\boldsymbol{A}$ and B depends on the number of time steps used to infer the model. This fact is not observed in Fig. 5.5 and Fig. 5.6. For 120 and 80 nodes, the MSE increases significantly when the number of time steps is lower than 60-time steps. For 40 and 20 nodes there is a growth of the MSE, but it is smaller when compared with 80 and 120 nodes. This behavior is observed because the estimate of hyperparameters depends more on the number of nodes respect to the number of time steps, and the MSE for 40 and 20 nodes for 100 time steps is larger respect to the MSE for 120 and 80 nodes.

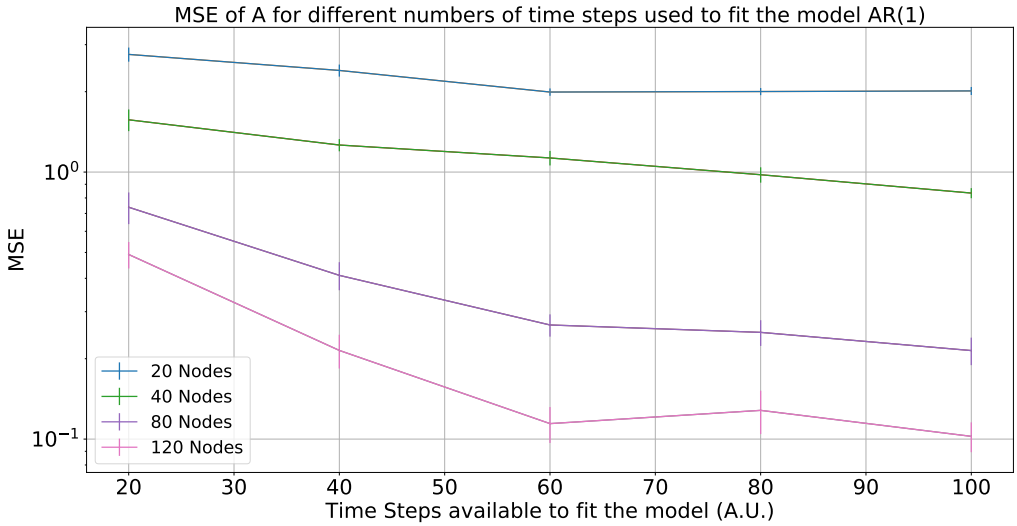

Figure 5.7: Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes

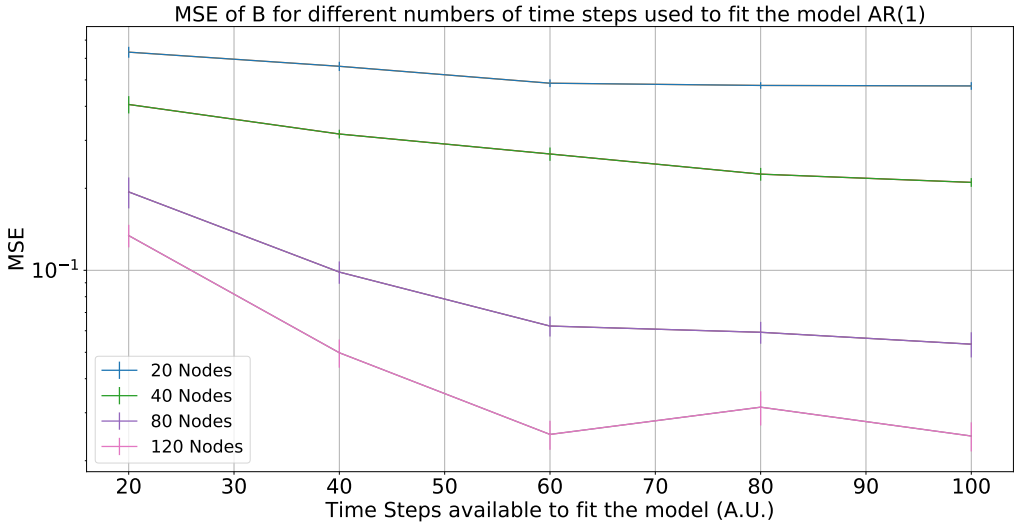

Figure 5.8: Evolution of MSE of B for different numbers of nodes

For Γ^t, we do not observe some specific patterns in the results of the model estimation. We can also observe that MSE for 20 and 40 nodes decreases: this shows that the model is not able to infer the hyperparameters in these configurations, as also suggested by the high values of MSE of $\boldsymbol{A}$ and B obtained for 20 and 40 nodes.

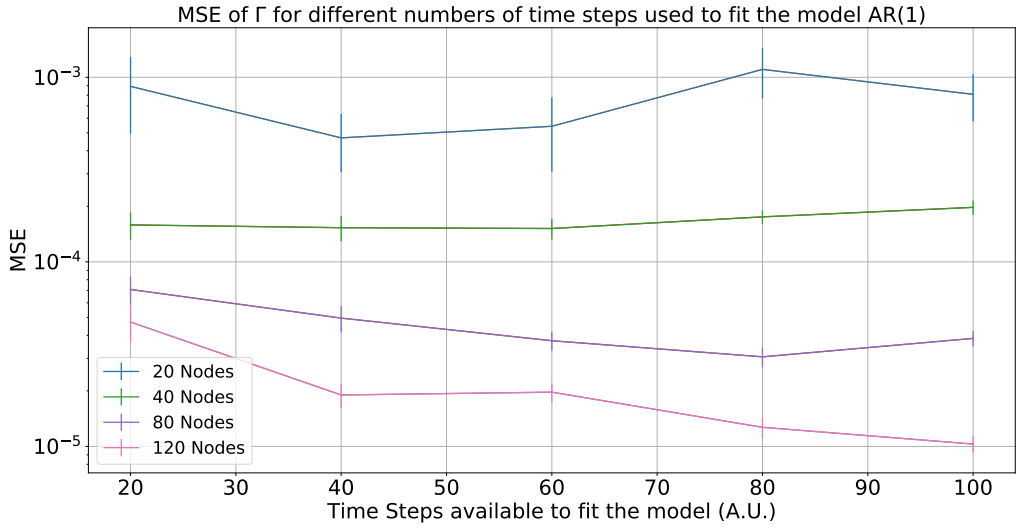

Figure 5.9: Evolution of MSE of Γ for different numbers of nodes

5.4 VAR-SBM Model Results

The VAR-SBM model generalizes further the AR-SBM model by capturing the dependence structure among the state variables of the network dynamics. Thus, we can compare the results obtained in AR-SBM Model with the results obtained with this model. Since this model is a generalization of the AR-SBM model, we can generate a network with the AR-SBM generative model and try to infer its parameters with the VAR-SBM model: we expect, in this case, that the estimated off-diagonal entries of B and Γ^t are very close to zero, since in AR-SBM model these parameters are null.
The model conserves the properties of stationarity, so the generative model is quite similar to the AR-SBM model.

5.4.1 Generative model

Since we build this model to compare its performance with the Xu and Hero model, we analyze only the VAR-SBM model with a hyperparameters structure similar to the Xu and Hero model. The matrices B and Γ are generated correlating the entries of θ that share the same row or column. So, for 2 groups, the matrices are like:

$$B = \begin{pmatrix} B_{diag} & B_{snb} & B_{snb} & 0 \\ B_{snb} & B_{diag} & 0 & B_{snb} \\ B_{snb} & 0 & B_{diag} & B_{snb} \\ 0 & B_{snb} & B_{snb} & B_{diag} \end{pmatrix}$$

$$\Gamma = \begin{pmatrix} \Gamma_{diag} & \Gamma_{snb} & \Gamma_{snb} & 0 \\ \Gamma_{snb} & \Gamma_{diag} & 0 & \Gamma_{snb} \\ \Gamma_{snb} & 0 & \Gamma_{diag} & \Gamma_{snb} \\ 0 & \Gamma_{snb} & \Gamma_{snb} & \Gamma_{diag} \end{pmatrix}$$

We have already observed what happens when we vary the entries of Γ, because its effect is the same as in AR-SBM Model. The main difference is in the coefficients of the matrix B. Here we consider $\boldsymbol{A} = \mathbf{0}$.

In Fig. 5.10, we observe the evolution of $\boldsymbol{\psi}^t$ and θ for $s_{diag} = 0.80$ and $s_{snb} = 0.05$.

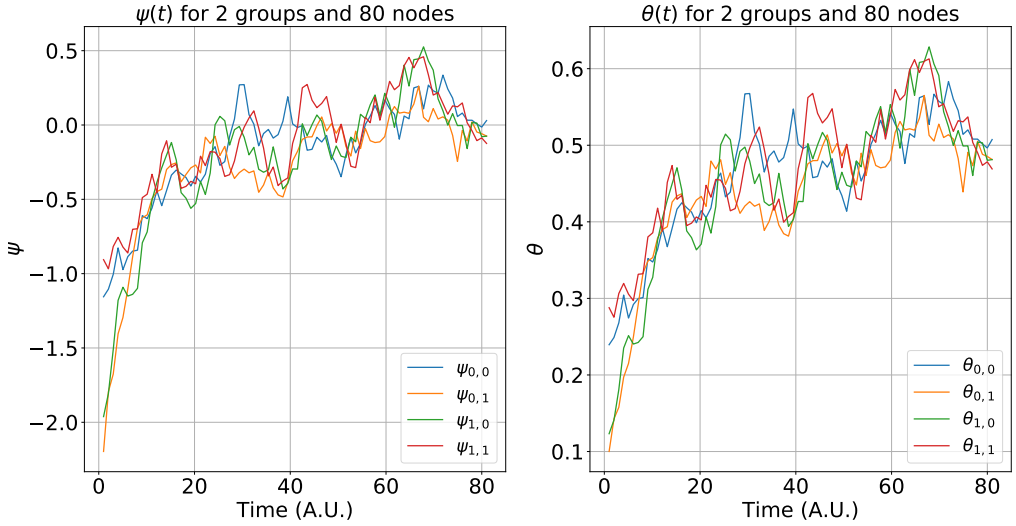

Figure 5.10: Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.80$ and $s_{snb} = 0.05$

If $s_{diag} = 0.50$ and $s_{snb} = 0.05$, the components of $\boldsymbol{\psi}^t$ tend to zero with a shorter transient phase, as we can see in Fig. 5.11.

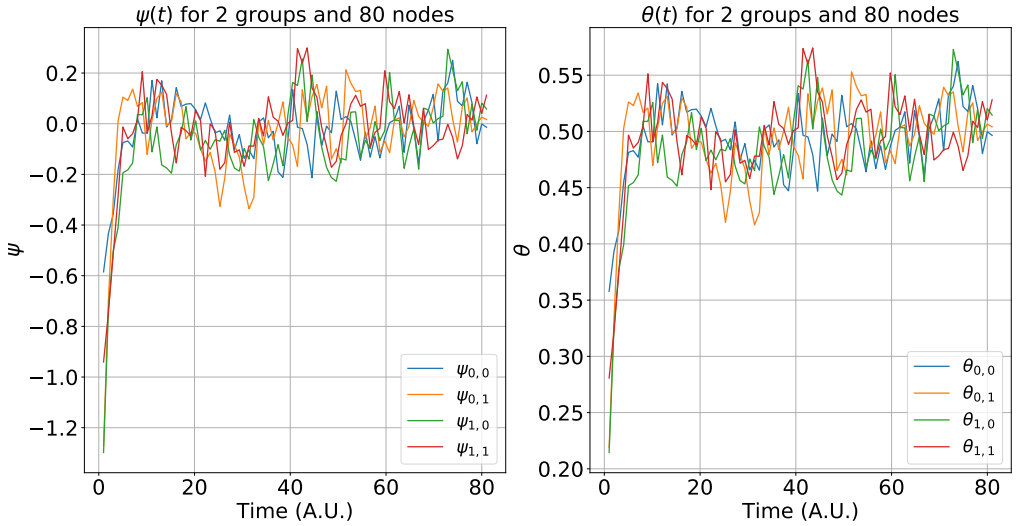

Figure 5.11: Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.50$ and $s_{snb} = 0.05$

For $s_{diag} = 0.20$ and $s_{snb} = 0.05$ (Fig. 5.12) and for $s_{diag} = 0$ and $s_{snb} = 0.05$ (Fig. 5.13), the components only fluctuate around the mean with a large variance.

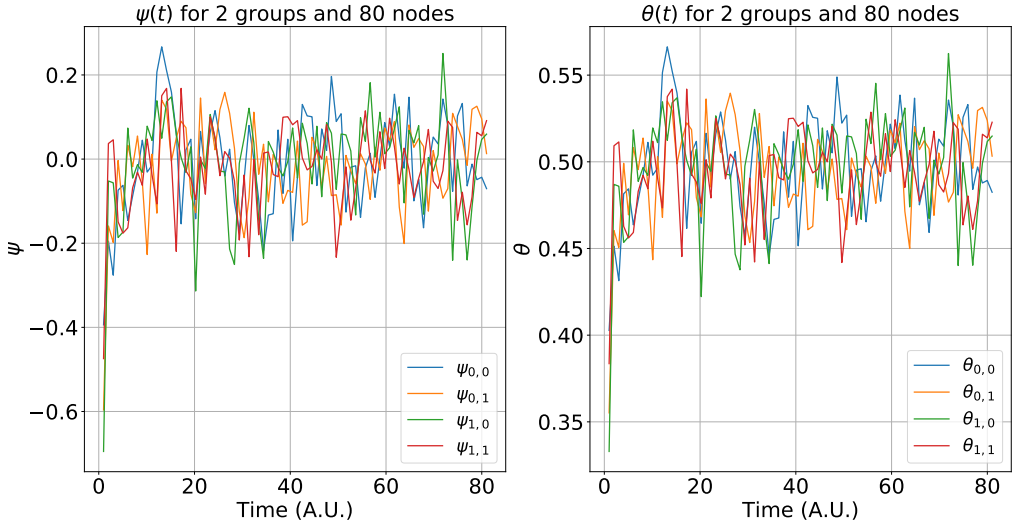

Figure 5.12: Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.20$ and $s_{snb} = 0.05$

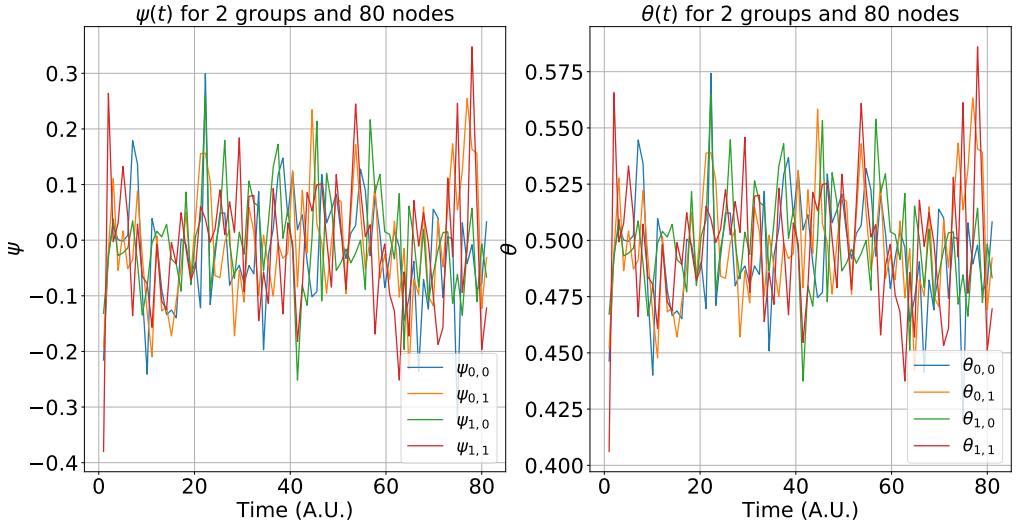

Figure 5.13: Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0$ and $s_{snb} = 0.05$

5.4.2 Inference of model's hyperparameters

Similarly to the case of the AR-SBM model, we investigate the inference of the hyperparameters of the VAR-SBM model. Also in this model, we collect 30 simulations, as done for the AR-

SBM model. To compare the three models, we use ARI, MSE e MSPE as metrics.
We immediately find out that VAR-SBM model depends on a large number of hyperparameters. In fact, for an AR-SBM model with k groups of nodes, we have to infer the only k A_i terms and k entries of the matrices B and Γ. In VAR-SBM model, if we have k groups of nodes, we have k A_i terms and k^2 entries of B and Γ. Hence, the model requires long time series to obtain a good estimate of the hyperparameters.

ARI

As for the AR-SBM model, we do not observe empirically variations of the ARI as a function of the number of time steps. On the contrary, it depends on the number of nodes used to simulate the network.

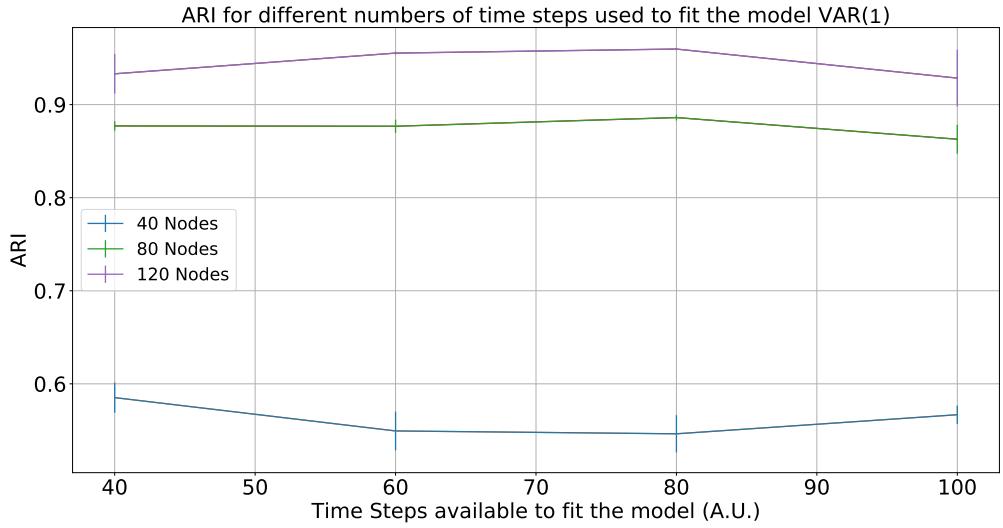

Figure 5.14: Evolution of ARI for different numbers of nodes in VAR-SBM model

MSE of ψ^t

Fig. 5.15 shows the dependence of MSE from the number of time steps. Some data have a large error bar, because, within the 30 simulations, one or two of them completely miss the inference of the hyperparameters. This output is observed at least once every 30 simulations, especially if we simulate a network with few nodes.

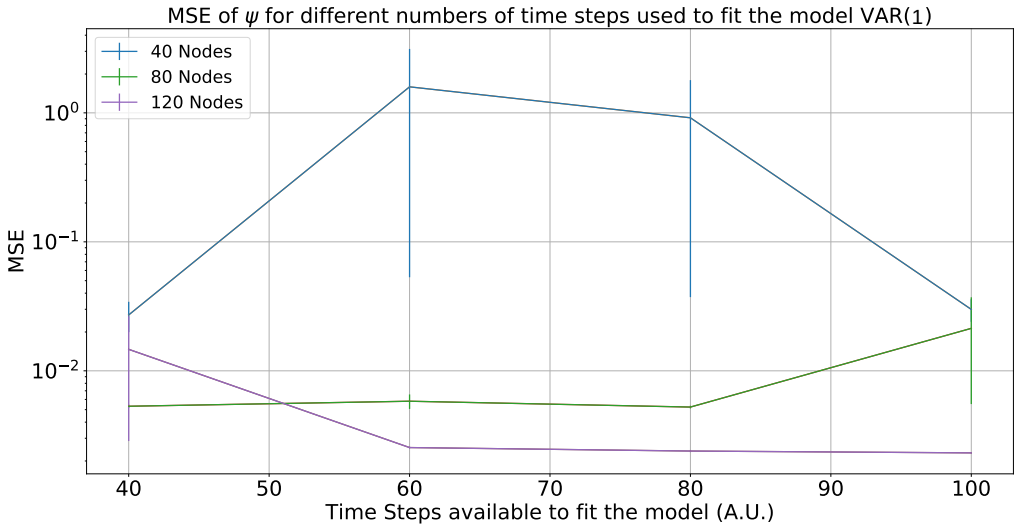

Figure 5.15: Evolution of MSE of ψ^t for different numbers of nodes in VAR-SBM model

MSE of A, B, and Γ

As shown in Fig. 5.16 and 5.17, A and B are dependent on the number of time steps used to infer the hyperparameters: the MSE decreases for a larger number of time steps. This fact is verified for every number of nodes used to configure the network. The dependence is not much evident for Γ, which depends more on the number of nodes.

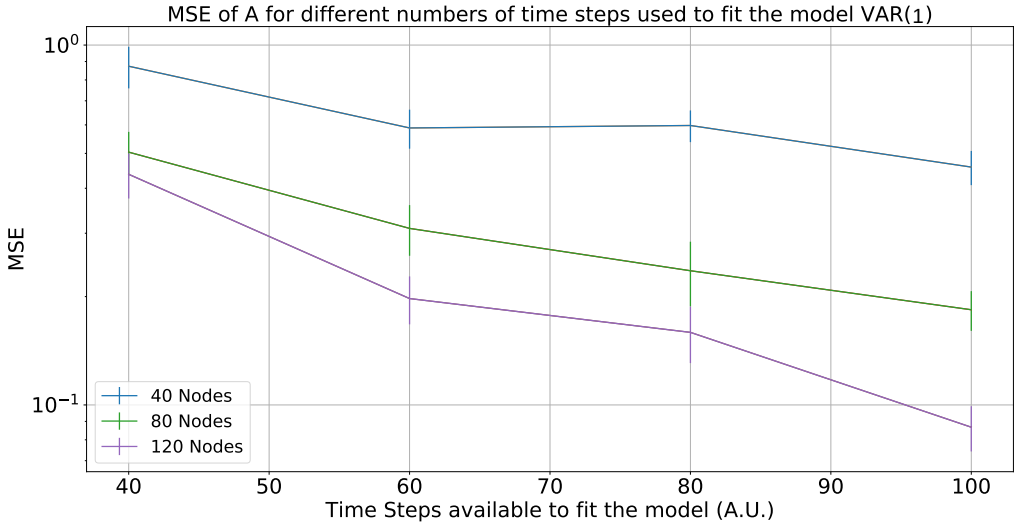

Figure 5.16: Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes in VAR-SBM model

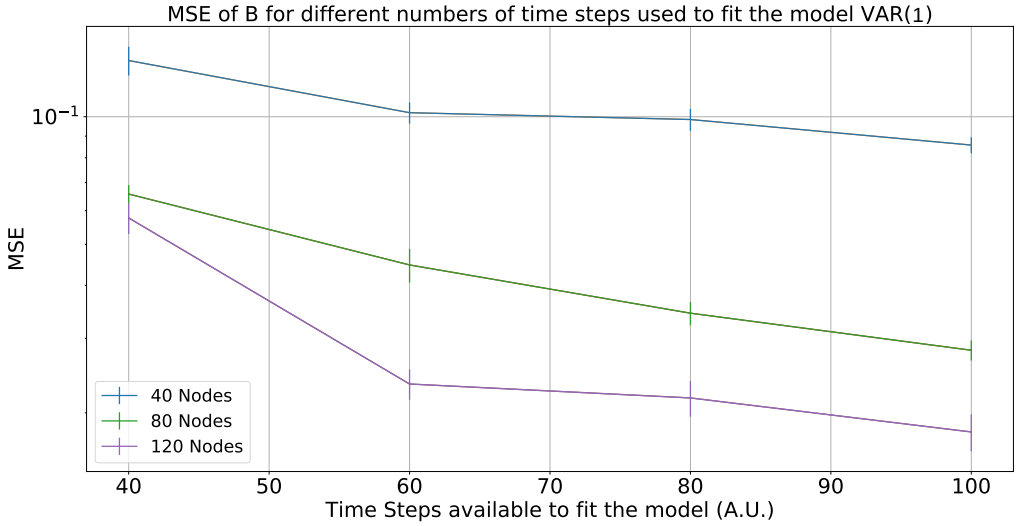

Figure 5.17: Evolution of MSE of B for different numbers of nodes in VAR-SBM model

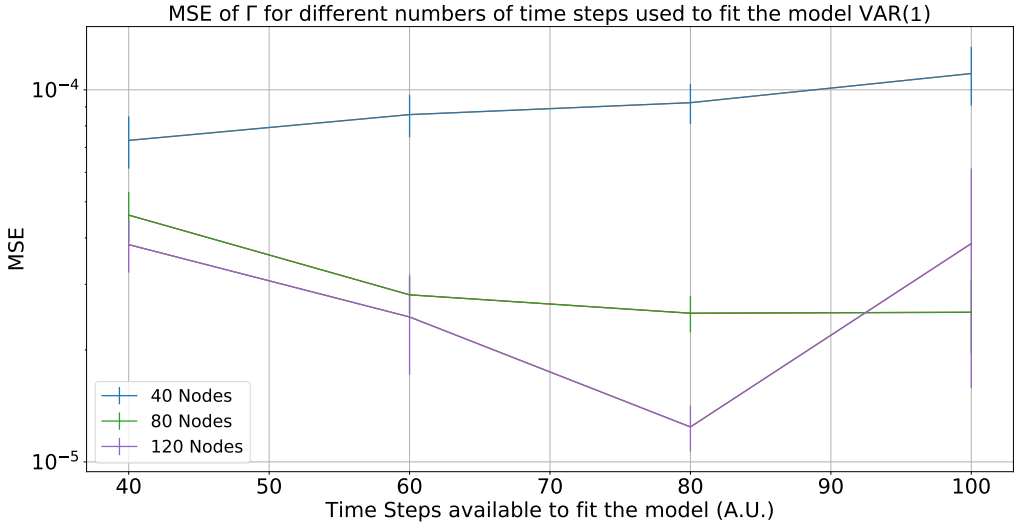

Figure 5.18: Evolution of MSE of Γ for different numbers of nodes in VAR-SBM model

5.5 Comparison between AR-SBM and VAR-SBM Model

Since the VAR-SBM model is a generalization of the AR-SBM model, we can compare the results obtained by the latter with those obtained by the VAR-SBM model. If we employ the two models on a network with k groups, the two models have a different number of parameters to infer: AR-SBM has to infer $3k$ parameters for $\boldsymbol{A}$, B, and Γ, while VAR-SBM has to infer $k + 2k^2$ parameters.

Here we simulate a network with the parameters used in Eq. 5.19, varying the number of time steps in the range $40 \leq N \leq 100$. For each number of time steps, we simulate 30 networks.

We aim to investigate what is the dependence structure for the state variables we measure according to the VAR-SBM model, when the network is generated according to the AR-SBM model, where no dependence structure is considered. In other words, how much the misspecification of the model affects the estimation results.

ARI and MSE of ψ^t

As we observe in Fig. 5.19 and 5.20, the two models obtain the same scores using both the metrics, even if the VAR-SBM model performs a little better than AR-SBM model.

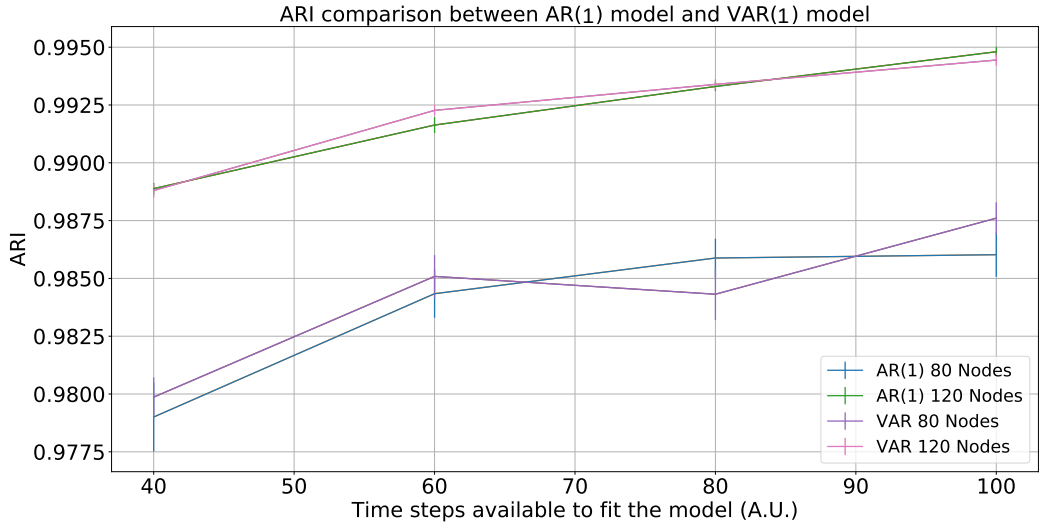

Figure 5.19: Evolution of ARI for different numbers of nodes in AR-SBM model and VAR-SBM model

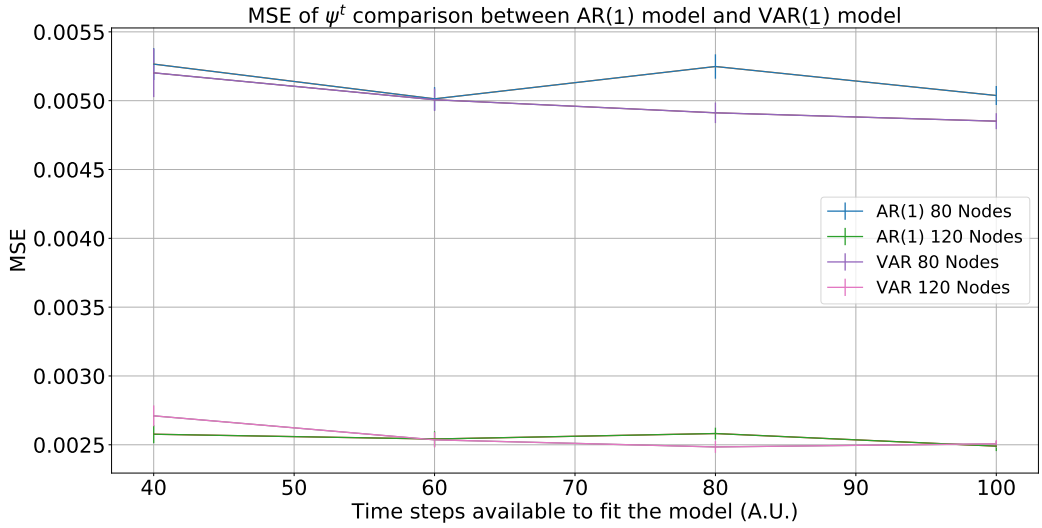

Figure 5.20: Evolution of MSE of $\boldsymbol{\psi}^t$ for different numbers of nodes in AR-SBM model and VAR-SBM model

5.5.1 MSE of A, B, and Γ

The analysis for MSE of $\boldsymbol{A}$, B and Γ gives different results. The VAR-SBM model performs always worse than the AR-SBM model, even if we compare the results for different numbers

of nodes, as we can see in Fig. 5.21. Vice versa, VAR-SBM model is more efficient in the inference of B and both the simulations for 120 and 80 nodes obtain the same score obtained by AR-SBM model for 120 nodes. In conclusion, the inference of Γ (generated with the AR-SBM model) is better with the VAR-SBM model than with the AR-SBM model. This results is quite surprising and it probably related to a finite-size effect due to the more flexibility of the VAR-SBM model respect to the AR-SBM model.

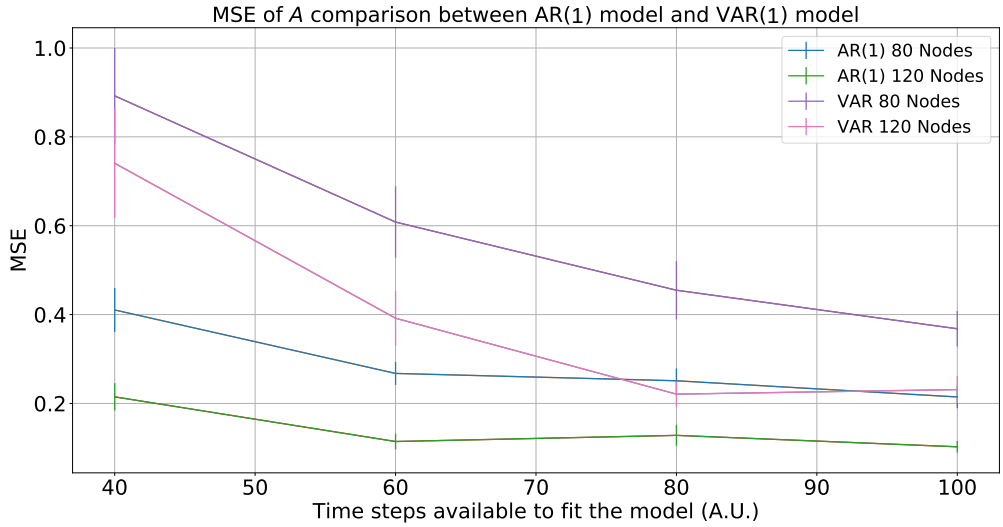

Figure 5.21: Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes in AR-SBM model and VAR-SBM model

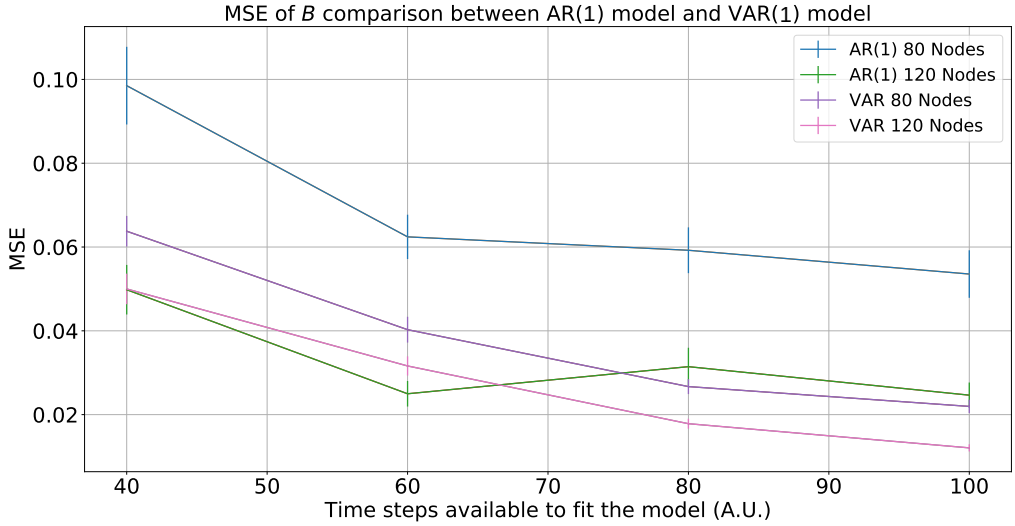

Figure 5.22: Evolution of MSE of B for different numbers of nodes in AR-SBM model and VAR-SBM model

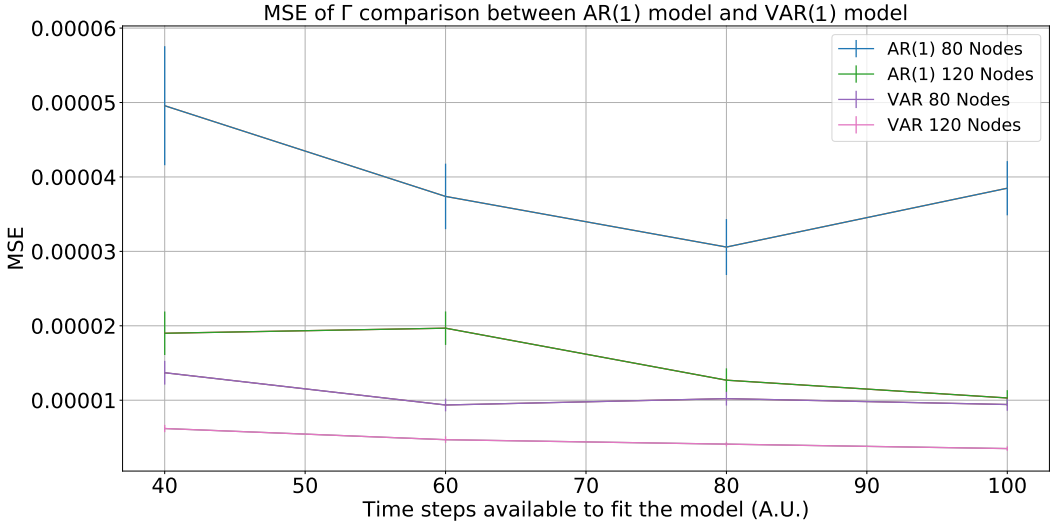

Figure 5.23: Evolution of MSE of Γ for different numbers of nodes in AR-SBM model and VAR-SBM model

5.6 Conclusion

Both the models AR-SBM and VAR-SBM are generalizations of the Xu and Hero model. These generalizations solve the issues related to non stationarity in the Xu and Hero model:

- The AR-SBM Model solves the problem of the absorbing states and introduces the mean reversion, but its state variables are uncorrelated;
- The VAR-SBM Model introduces a dependence structure for the state variables, which is used also in Xu and Hero model;

However, these two models required a large number of time steps to infer the hyperparameters of a network, while Xu and Hero model depends only on the previous step: this means that AR-SBM and VAR-SBM are not competitive for short temporal networks and their inference is associated with larger estimation error under these conditions.

Chapter 6

Application to Interbank Network

In this chapter, we apply the Xu and Hero model and the VAR-SBM model to a real dynamic network to filter the underlying structure of the graph. The dynamic network is generated using a dataset which describes the e-MID transactions in the period 2009-2015. For this reason, we first review the literature of the interbank market and of the e-MID to have an suitable background to compare the results obtained.
The chapter is divided into three sections:

1. The first part concerns the study of the interbank market. We discuss the main features of the interbank network analyzed over the years and we focus on the e-MID network, which the basis of our numerical simulations;

2. In the second part, we clarify the rules and criteria used to create the dynamic network from the dataset and we expose which measures are used to validate the inference process;

3. The third part concerns the results of the inference process on the dynamic network. Our analysis will focus on the inferred structure of the network and on the comparison of our results with Barucca and Lillo [5].

The first section and part of the second one are based on several scientific works which describe the structure of the interbank market and of the e-MID network. All the numerical results presented in this chapter are original, except for the analysis of the e-MID in Section 6.2, based on the paper of Barucca and Lillo [5].

6.1 Interbank Network and the e-MID

The interbank market is the global network utilized by banks to lend funds each other. The recent financial crisis has brought a lot of attention on the interbank market. In fact, many mathematical models were proposed through these years with two main aims: first to describe the structure of the interbank market and to predict further changes in the network, second

to have several tools «to develop early warning indicators of systemic risk» [24]. G. Iori describes the strict relation between the interbank market and the network analysis:"the global financial crisis of 2008 has stressed the importance of market interconnectedness on financial stability and highlighted the need to analyse financial markets as a system of individually complex institutions that are connected to one another in a complex network of counterparty exposures" [24]. One of first analysis of the interbank market is "The network topology of the interbank market" [7] developed for the Oesterreichische Nationalbank in 2003. Starting from this milestone work, other scientific papers have focused on the study of the network structure of the interbank market, in particular:

- "Fractal network derived from banking trasaction - an analysis of network structures formed by financial institutions" [23], developed for the Central bank of Japan;
- "The topology of Interbank Payment Flows, Federal Reserve Bank of New York - Staff Reports" [3], developed in the Fed;
- "The network topology of CHAPS Sterling" [6], developer for the Bank of England;
- "The topology of Danish Interbank Money Flows" [35], develop by the Finance Research Unit of the Department of Economics of the Copenhagen University.

Different studies have pointed out that the interbank network displays some regularities: very low connectivity, an heterogeneous degree distribution, low average distance between nodes, disassortative mixing, small clustering, and an heterogeneous level of reciprocity [7] [26] [20] [12]. Furthermore, the interbank market is a multilayer network where several kind of credit relations are present concurrently [4]. Usually, when analyzing a single layer of the interbank market, the most frequent inferred structure is the core-periphery, as shown in literature [20] [19] [25] [26].
In the next subsection we study the e-MID by analyzing its community structure by means of the dynamic stochastic block models previously introduced.

6.1.1 The e-MID network

The e-MID is the Italian electronic market for the interbank deposit. It «is a screen-based platform for trading of unsecured money-market deposits operating in Milan» [5]. We aim to investigate the structure of this network using the stochastic block models we introduced in the previous chapters. To achieve this goal, we decide to study and to expose the context and the main events concerning the e-MID in the period which cover our analyzed time window (2009-2015). This time window includes with the Eurozone sovereign debt crisis. The European Central Bank (ECB) decided to apply exceptional measures as reaction to the crisis, one of the most important being the LTROs (Long Term Refinancing Operations), which took place on the 22nd of December 2011 and on the 29th of February 2012. These measures represent in

our analysis the expected event in the time series builds with the transaction in the e-MID in the investigated period. The LTROs are so important because the ECB employed 1.03 trillion of Euro and the Italian banks took 110 billion Euro [5]. Thus, the e-MID banks started to receive funds from an external institution, instead of only exchange money between themselves. For this reason, our study aims to investigate if the LTROs had an effect on the e-MID network and its structure and, if any, to infer the change in the network's structure. Other important measures were applied in the studied period:

- the Securities Market Program (SMP) on May 10, 2010 and on August 2, 2011;
- the Outright Monetary Transactions (OMT) on August 2, 2012;
- In the second week of July of 2012, the ECB lowered significantly the interest rates.

In the following section, we review the features of the e-MID network, in particular by focusing on the work of Barucca and Lillo [5]. In Section 6.3 we expose the results of our numerical simulations.

6.2 Dataset

The dataset contains the e-MID transactions from July 2009 to February 2015 and we focus on the overnight exposures. Our analysis focuses on the Italian banks, which constitute 68% of the banks in the whole dataset and 84% of the total volume in trading in this period [5].
The dataset contains five different information for each transaction:

1. Two codes which identify the borrower and the lender in the transaction;
2. The starting date and the ending date of the transaction;
3. The amount of the transaction in Euro.

Following the results obtain by Barucca and Lillo [5], we analyze the macro-information of the dataset. In Fig. 6.1, we observe the evolution of the number of banks in time and the corresponding trading volume. A key variable employed in the analysis is the time scale of aggregation of the transaction [5]. In fact, the dataset contains data for each day of the studied time window, but the inferred structure of the network could change for different time scales of aggregation. In the case of long time scales, it is possible to identify long run tendencies in the structure of the bank's credit relations [5]. The data in the Fig. 6.1 and Fig. 6.2 are grouped by week. The first remark is that the LTROs strongly affected the number of banks in the e-MID and the same for the traded volume, since both the numbers collapses drastically after the LTROs. The foreign banks start decreasing in number during the second half of 2011, and, after the first LTRO, their number becomes a small part of the observed banks.

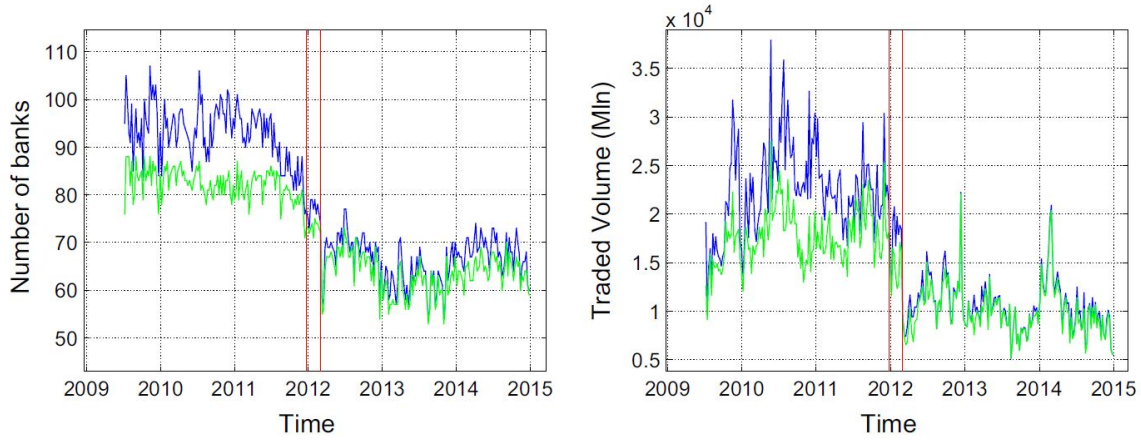

Figure 6.1: Weekly number of banks (left) and weekly traded volume in Million of Euros (right) traded ate-MID from June 2009 to December 2014. The blue line refers to all the banks while the green line refers only to Italian banks. Vertical red lines indicate the two LTRO measures by ECB on Dec. 22, 2011 and Feb. 29, 2012 [5]

The structure of the network, on the other hand, is not necessarily affected by the LTROs. Barucca and Lillo [5] observe that « the density of the network (Fig.6.2, left panel), does not display a similar transition.» The fact implies that the remaining banks in the e-MID did not create new links with banks they were not connected to. Barucca and Lillo suggest two possible explanations for this phenomenon:

- Creation of new opportunities outside the e-MID;
- Increasing in the traded amount in the e-MID with the banks they were still trading before the LTROs.

The right panel in Fig. 6.2 shows the time series of the average amount exchanged per link, while the left panel shows the density of the network, which is the number of observed links over the number of possible links in the network. The right panel in Fig. 6.2 does not clarify which alternative is the correct one, since «the time series does not show a significant increase around the time of the LTROs, indicating that remaining banks did not increase the volume of trading with the remaining banks, but either trade less in the interbank market or trade outside e-MID» [5]. Thus, is it not possible, with this dataset, to distinguish between the two alternatives.

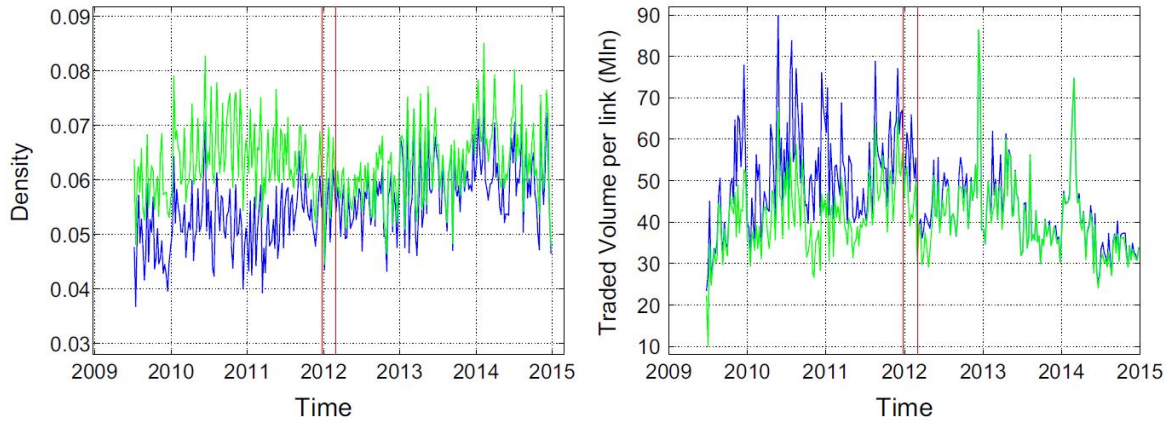

Figure 6.2: Left. Weekly density (i.e. the number of links divided by the total number of possible links) of e-MID from June 2009 to December 2014. Right. Weekly volume per link (i.e. total volume in Million of Euros divided by the number of links. The blue line refers to all the banks while the green line refers only to Italian banks. Vertical red lines indicate the two LTRO measures by ECB on Dec. 22, 2011 and Feb. 29, 2012 [5]

6.2.1 Criteria applied to the dataset

In our analysis, we define several criteria to generate the dynamic network from the dataset we have. We list them one by one:

1. Time scale: we decide to aggregate all the transactions to a weekly time scale. This means to sum all the transactions which start and end in each time window equivalent to the time scale. In Barucca and Lillo [5] three different choices for the time scale are explored (daily, weekly and monthly): we limit our analysis to the weekly time scale. The case of monthly time scale is not considered for reasons of low dimensionality of the time series of network snapshots. The analyzed period covers more or less 5 years, which correspond to 81 months and 320 weeks (each month corresponds to 20 working days and each week to 5 working days). Since we have imposed the stationarity condition on our models, and since we expect the LTROs could correspond to a change in the structure of the dynamic network, we decide to employ the two models on two different choices of periods: the Xu and Hero model is applied to the whole studied period (2009-2015), while for the VAR-SBM model we decide to split the dataset in two distinct periods. The first period is the one pre-LTROs (July 2009-December 2011) and the second period the one after the LTROs (January 2012-June 2015). A daily time scale is not considered since it requires a long computational time and it is out of our scope.

2. Bank selection: the dataset contains 166 different bank identity codes, but most of them have few relations (links) at each time step with other banks. This implies that the

adjacency matrices, both weighted and unweighted, would be filled with zero entries for rows and columns related to these banks. For this reason, restrict our analysis to the banks which have operated in the market at least 2000 transactions in the considered time period. Thus, the number of studied banks is reduced to 68, instead of 162. All the 68 banks are Italian.

3. Weighted or unweighted network: in order to study the community structure of the e-MID network by means of the previously described stochastic block models, we built an unweighted dynamic network. In a possible further analysis, we could generalize the algorithms to the weighted version and observe the difference with the unweighted one. The unweighted network is created following the criteria defined in Barucca e Lillo [5]: if in the considered time scale the amount exchanged between the two banks is greater than zero, we fill the adjacency matrix entry with the value "1", otherwise the value is zero.

4. Directed or undirected network: we decide to follow the same assertion used in Barucca and Lillo [5]: « passing from a directed to an undirected network, information about the structure is lost and structure identification may be strongly influenced by total degree». Thus, we create directed network that links the lender to the borrower. Further analysis could be the comparison between the directed and undirected network.

5. The number of groups: in our analysis, we consider only the case of two groups, since we do not any comparison result in literature to justify a different choice.

The result of these criteria is an unweighted, directed dynamic network. We decide also to exclude the AR-SBM model from the analysis of the dataset and only the VAR-SBM model is employed to infer the structure of the dynamic network. This choice is supported by the comparative analysis between the two models shown in Section 5.5, which shows that the VAR-SBM model obtains better performance respect to the other model. Furthermore, the VAR process is able to describe a large variety of time series which the AR process cannot. Hence, the VAR-SBM model is more general respect to the AR-SBM model.

6.2.2 Inference Validation

We need to validate the numerical results obtained by the inference process. We use two metrics to do this: the Mean Square Error (MSE) and the Cosine Similarity. Since we do not have a real measure of hyperparameters of the network, we must use some assumption to compare our inferred hyperparameters with the observed data. The basis of the validation is the observed number of links at each time step. This data depends only criteria used to construct the dynamic network and it does not depends on the inference process. For this reason, we decide to compare the predicted numbers of links in the network with the observed numbers of links in the adjacency matrix. On the other hand, we have to group the observed number of

links using the population of the two groups at each time step, which is not an information given or observable, but it can only be the result of the inference process. Anyway, by considering the analysis of the numerical simulations of both the models we employ (see section 4.3.2 and 5.4.2), we observe that both the models obtain excellent performance in community inference. Thus, we assume that the inferred groups are correct: this choice allow us to group the observed number of links in a non-observed manner and to consider this still correct.

MSE

We adopt the following algorithm to assess what is the expected number of links in each group:

1. For a Random Walk process (as the Xu and Hero model), the best-predicted state is the previous state, so: $\hat{\boldsymbol{\psi}}^{t+1|t} = \boldsymbol{\psi}^{t|t}$
2. Using the logistic function, we obtain the estimated affinity matrix $\hat{\Theta}^{t+1|t}$
3. Using the label switching method, we infer the label $\hat{\boldsymbol{c}}^{t+1}$
4. Combining $\hat{\Theta}^{t+1|t}$ and $\hat{\boldsymbol{c}}^{t+1}$, we calculate the predicted number of links for each group $\hat{\boldsymbol{k}}^{t+1}$ in the adjacency matrix W^{t+1} at time $t+1$
5. Using $\hat{\boldsymbol{c}}^{t+1}$, we group the links in the adjacency matrix W^{t+1} to obtain observed the number of observed links for each group $\boldsymbol{k}^{t+1}$
6. Finally, $MSE = \frac{1}{N_k}\sqrt{||\boldsymbol{k}^{t+1} - \hat{\boldsymbol{k}}^{t+1}||^2}$

Where N_k is the number of components of $\boldsymbol{k}^{t+1}$, equal to 4 in the case of 2 groups.

Cosine Similarity

The cosine similarity is another kind of metric, that is defined as follows:

$$\cos\theta = \frac{\boldsymbol{A}\cdot\boldsymbol{B}}{||A||||B||} \tag{6.1}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are vectors. In our case, the entries of the vectors are the number of links for each component of the affinity matrix. The first vector is for the observed number of links and the second one is for the predicted number of links.

6.3 Result

In this section, we analyze the results obtained in the numerical simulations. This section is divided into three subsections:

- Xu and Hero model and results' validation;
- VAR-SBM model and results' validation;
- Comparison between the models.

Each subsection is split into multiple parts to focus on different aspects of the inference results: structure of the network and the affinity matrix, the evolution of the community structure of the banking network, analysis of the MSE, and cosine similarity.

6.3.1 Xu and Hero model

Structure of the network

The first aspect we consider is the structure of the dynamic network. We observe a bipartite structure of the network during the whole period covered by the dataset. As we see in Fig. 6.3, the relation between the entries of the affinity matrix is:

$$\theta_{0,1} > \theta_{0,0} > \theta_{1,1} > \theta_{1,0} \tag{6.2}$$

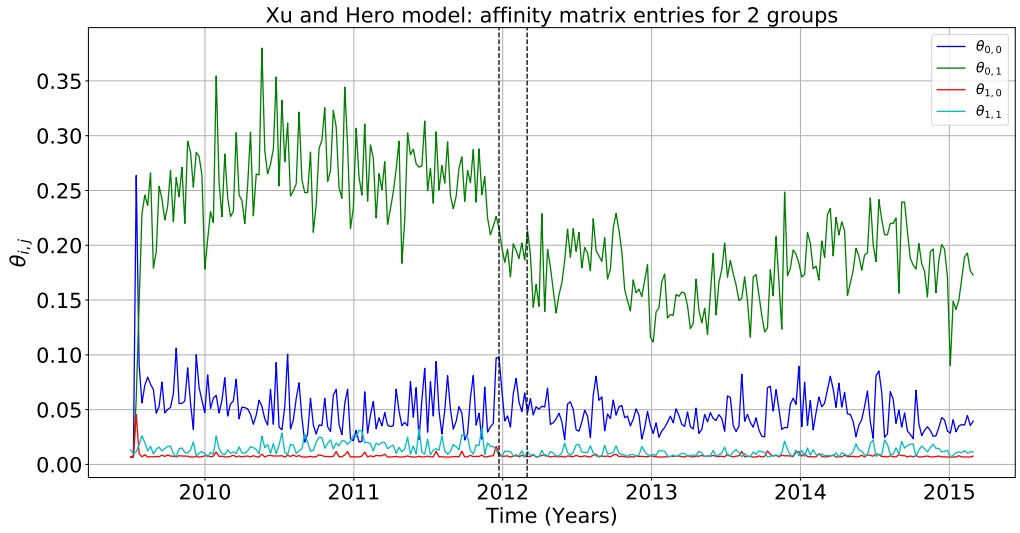

Figure 6.3: Entries of the inferred affinity matrix Θ^t

Since we identify the lender with the group number "0" and the borrower with group number "1", the previous equation could be rewritten:

$$\theta_{L,B} > \theta_{L,L} > \theta_{B,B} > \theta_{B,L} \tag{6.3}$$

This result is in line with the analysis obtained in Barucca e Lillo in [5] with a static model. Exploring deeply the entries of the affinity matrix Θ^t, we can observe that (see Fig. 6.4 and Fig. 6.5):

1. The entry $\theta_{L,B}$ decreases significantly after the LTROs: before the LTROs, we observe that $0.25 < \theta_{L,B} < 0.30$ for most of the weeks. After them, we observe that the observe entry has a lower value, especially in 2013, where $\theta_{L,B}$ and $\theta_{L,L}$ have very close values. This implies that the observed bipartite structure after the LTROs is still evident, but weaker.

2. The time series relative to $\theta_{L,L}$ is not affected by the LTROs. This probably implies that the lenders that remain in the e-MID after the LTROs continue to have transactions between them as they did before.

3. It is quite interesting to observe that $\theta_{B,B}$, in the period 2010-2012, has a mean value slightly different from zero, while after the LTROs, its value is more or less constant and almost close to zero. We can deduce that the few borrowers decided to start transactions between themselves in 2010-2012, and the LTROs completely modify their behavior.

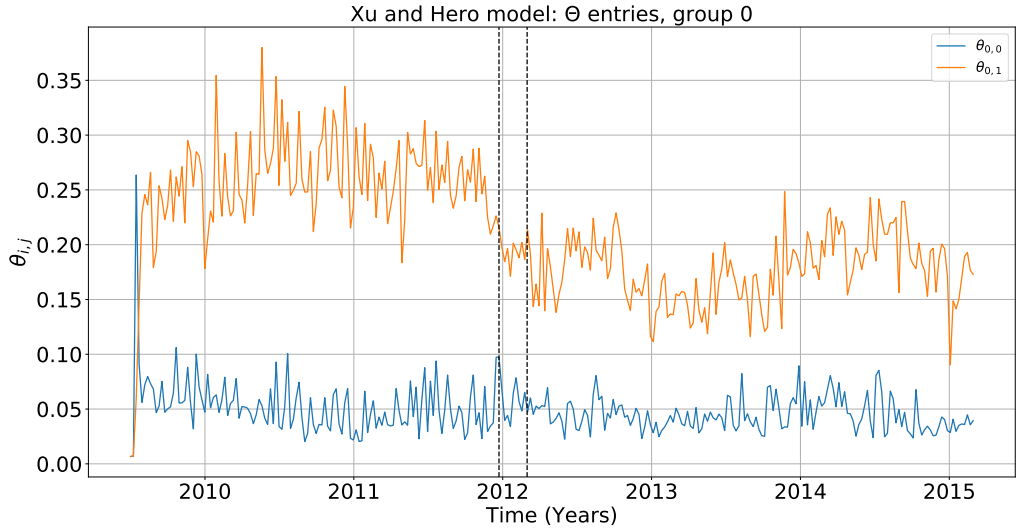

Figure 6.4: Entries of the inferred affinity matrix Θ^t

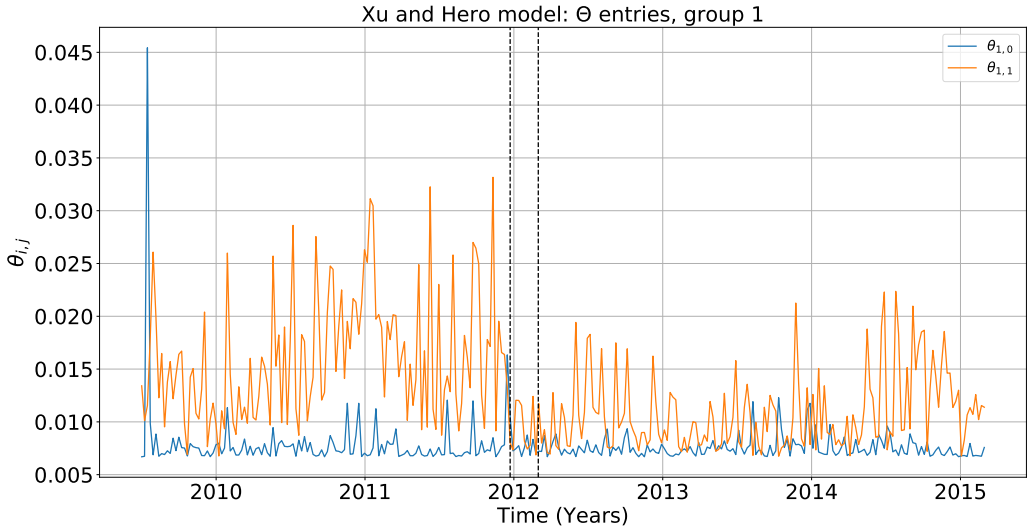

Figure 6.5: Entries of the inferred affinity matrix Θ^t

Evolution of the bank's behavior

We observe that the number of lenders and borrowers changes through time and we can divide the analysis into three different behaviors of the banks in the e-MID market (Fig. 6.6).

- The period Jun.2009-Oct.2010 and the period May.2014-Jun.2015 are characterized by a stable number of lenders and borrowers, composed of 27 ± 6 and 41 ± 6 banks respectively. These periods overlap those described in table 1 in Barucca e Lillo [5] where the bipartite structure is more likely to be observed, i.e. the periods corresponding to 2010-2011 and to 2014. Also the numbers of banks that belong to the group of lenders and borrowers overlap with their estimates: in fact, Barucca and Lillo [5] infer the number of 45 ± 9 banks in the group of borrowers and 30 ± 8 banks in the group of lenders.

- In the period Oct.2010-Dec.2011, the difference between the number of borrowers and lenders becomes very small. In this period we observe also that the number of lenders becomes slightly larger than the number of borrowers for a few weeks.

- In the period Jan.2012-Apr.2014 we observe a change in strategy for most of the banks: in fact, the number of borrowers becomes very large (we observe 54 banks in this group in Dec.2013).

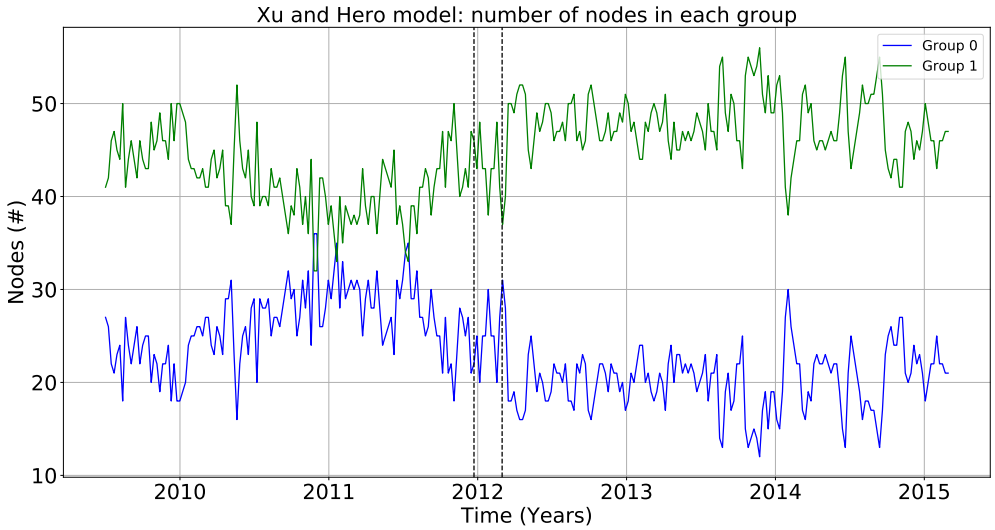

Figure 6.6: Number of banks in each group: the green line refers to the borrowers and the blue line refers to the lenders

We deduce that the LTROs had a strong impact on the banks' behavior, because many banks decided to change their strategy in e-MID and they started to borrow again.

Results' validation

As we describe in Subsection 6.2.2, we use two different metrics to validate our numerical results. Firstly we study MSE. In Fig. 6.7 we observe two data series: the first one is the number of the observed links in the network at each time step (orange line), the other one is the calculated MSE of the observed number of links, computed with the predicted number of links (blue line). In Fig. 6.8, we observe that the ratio point by point between the time series described in Fig. 6.7. We observe that the ratio oscillates in the range $0.02 - 0.10$ for most of the time steps (Fig. 6.8). In Fig. 6.8, we observe that the time step associated with the highest ratio is set in the middle of the LTROs, and it reaches the value of 0.22. We also note a time step with a high ratio at the end of the time series, but it corresponds to a week with very few observed links ($\simeq 100$). Thus, this value is more probably related to a lack of data with respect to a large estimation error. Hence, the estimate of the hyperparameters of the interbank network could be considered acceptable because the ratio is almost under 0.10.

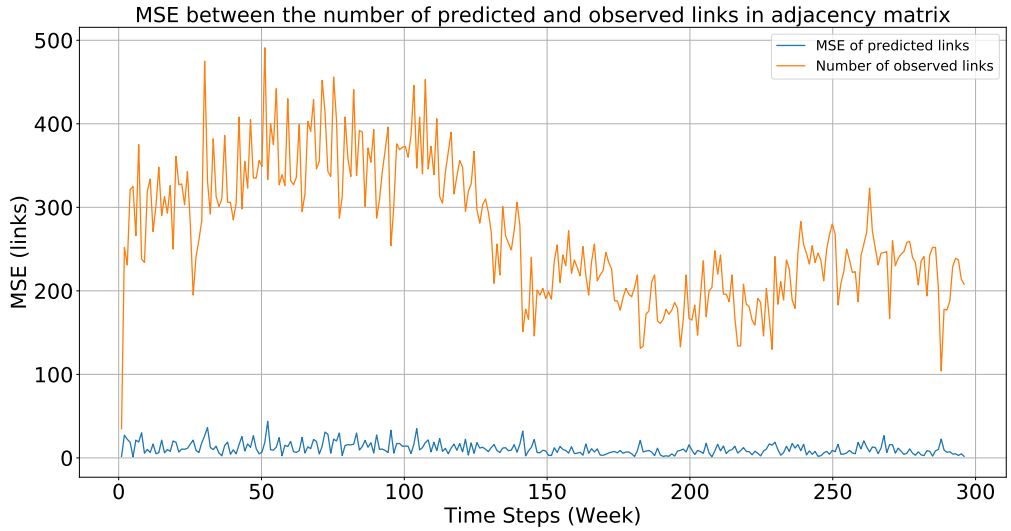

Figure 6.7: Analysis of the MSE: in orange the number of observed links in the network, in blue its MSE value at each time step.

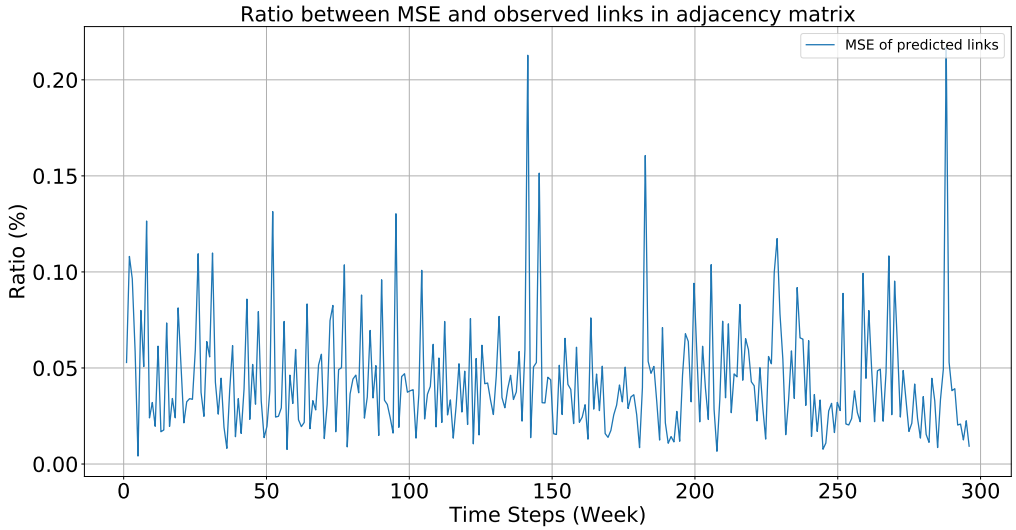

Figure 6.8: Analysis of the MSE: the ratio between the blue and the orange lines in figure 6.7

If we split the previous analysis for each entry of the affinity matrix Θ^t (Fig. 6.9), we can explore more details: we observe that, for the entries $\theta_{0,0}$, $\theta_{0,1}$, and $\theta_{1,1}$, the punctual difference between the prediction and the observation is not wide. For these three entries, we observe a good agreement between the predicted and the observed time series. The entry $\theta_{1,0}$ is not statistically interesting (we note that the maximum of the observed links for that component is

equal to 20 links). Even if the comparison is not completely satisfactory, we can say that the observed time series is always comparable with the predicted one, except for $\theta_{1,0}$.

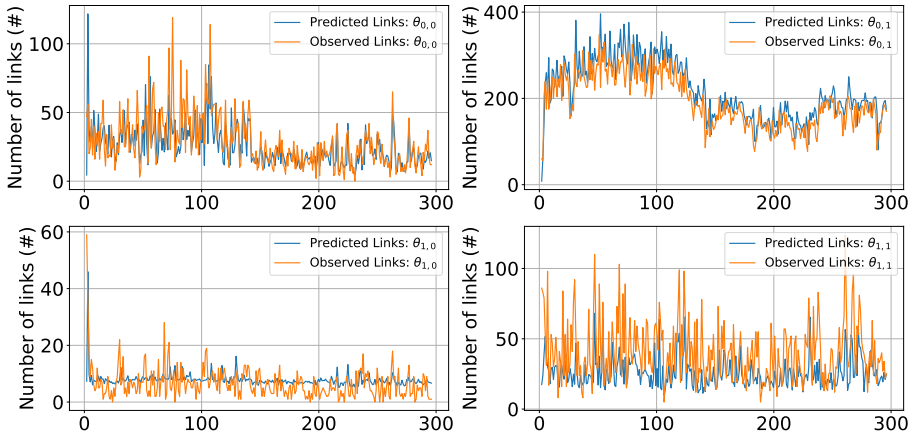

Figure 6.9: Evolution of the observed and predicted groups of links for each component of $\boldsymbol{\Theta}$

Concluding, in Fig. 6.10, we observe that $\cos\theta > 0.94$ for most of the time steps. Excluding the first time step, we note low scores in the second part of the time steps: this fact probably depends on the small difference between $\theta_{0,1}$ and $\theta_{1,1}$ in the investigated period, as we see in Fig.6.3.

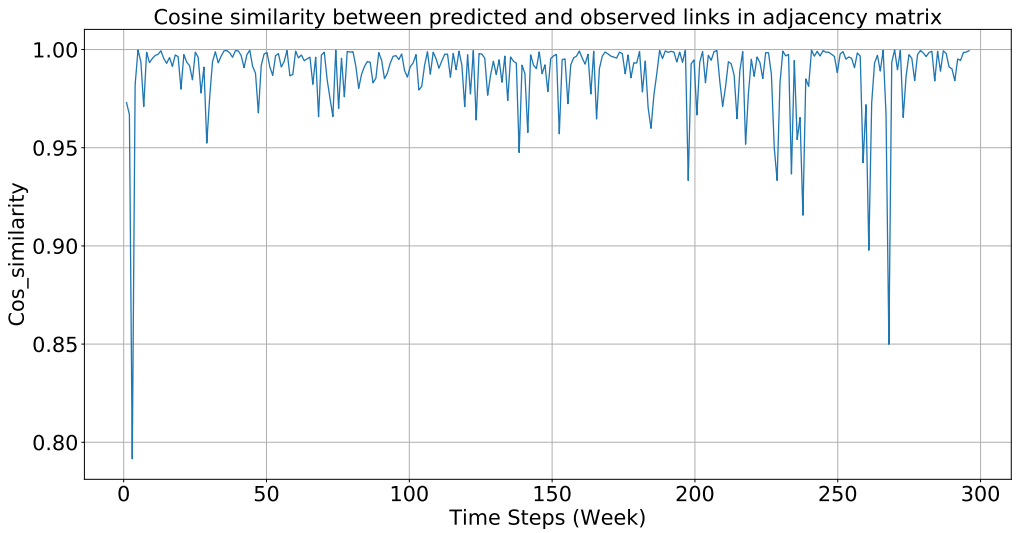

Figure 6.10: Cosine similarity for Xu and Hero model

We decide also to include in the inference validation the analysis of the fraction of banks

that change the group between each time step. This analysis is related to the hypothesis of Xu and Hero in [40]. They affirm that the algorithm works for dynamic networks with a small fraction of nodes that change group between time steps and, in their simulations, they use a percentage equal to 10%. In Fig. 6.11, we observe that the percentage inferred oscillates in the range $[2\% - 21\%]$, which is a reasonable range for the algorithm, as we also proved numerically in our analysis in Chapter 4.

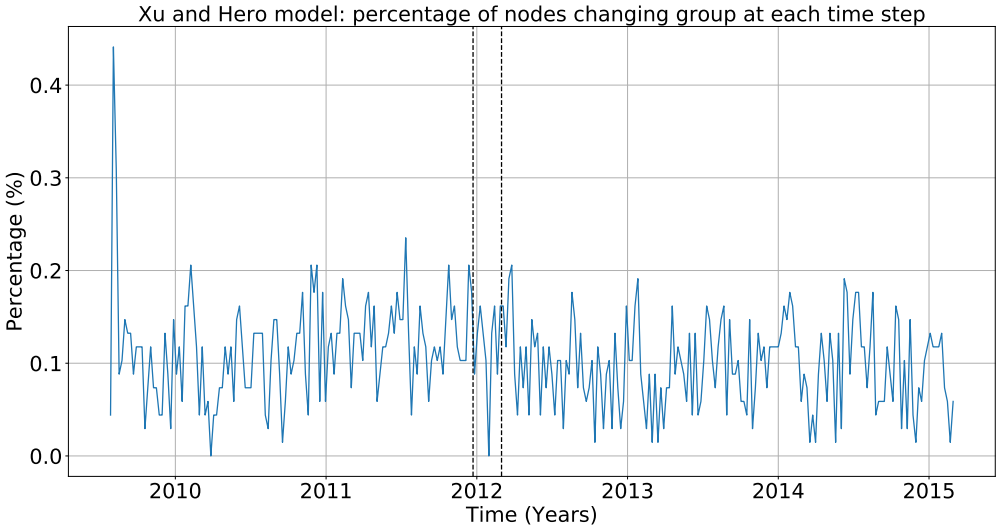

Figure 6.11: Fraction of banks that leaves their groups at each time step in Xu and Hero model

6.3.2 VAR-SBM Model

In this Section, we consider the case of the VAR-SBM model. We restrict our analysis only to this case because the VAR-SBM model generalizes the AR-SBM model by including the description of the dependence structure of the state variables.

The dataset is the same used for the Xu and Hero model and we analyze the same banks to compare the results. Since the VAR-SBM model is a stationary model, we cannot use this model to analyze the whole dataset at the same time, because we have supposed that the LTROs have an effect on the interbank network, making it non-stationary during the considered period. Hence, we split the dataset into two parts:

- A first period (June 2009-December 2011), that considers the period before the LTROs
- A second period (January 2012- June 2015), that considers the period after the LTROs

This division is due to the LTROs, which have occurred on Dec. 22, 2011 and Feb. 29, 2012.

The first remark is that the new model infer the same information of the Xu and Hero model, but as we will see in the following figures, the stationarity property mediates the dynamic of the time series, i.e. the local variations of the time series of the affinity matrix entries, observed mostly in the period after the LTROs for the Xu and Hero model (see subsection 6.3.1), are now flattened around the mean of the inferred process.

Structure of the network

In Fig. 6.12, 6.13, and 6.14 we observe the evolution of the entries of the affinity matrix Θ for the VAR-SBM model.
The first note is that the difference between $\theta_{L,B}$ and $\theta_{L,L}$ is constant in the two periods, even if it is reduced after the LTROs. Furthermore, we observe in Fig.6.14 that $\theta_{B,B}$ is now different from zero in the whole period pre-LTROs, while in the Xu and Hero inference process it is almost zero in the period Jun.2009-Oct.2010. This is the effect of the stationarity of the VAR process we have described before. The Eq. 6.3 between the entries of the affinity matrix we found for the Xu and Hero model still holds. This implies that the VAR-SBM model infers a bipartite structure as observed in the Xu and Hero model.

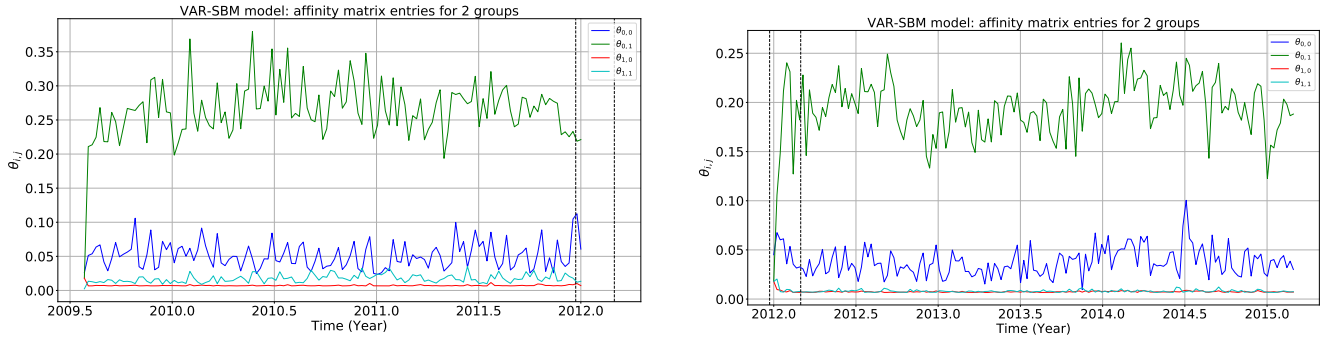

Figure 6.12: Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs

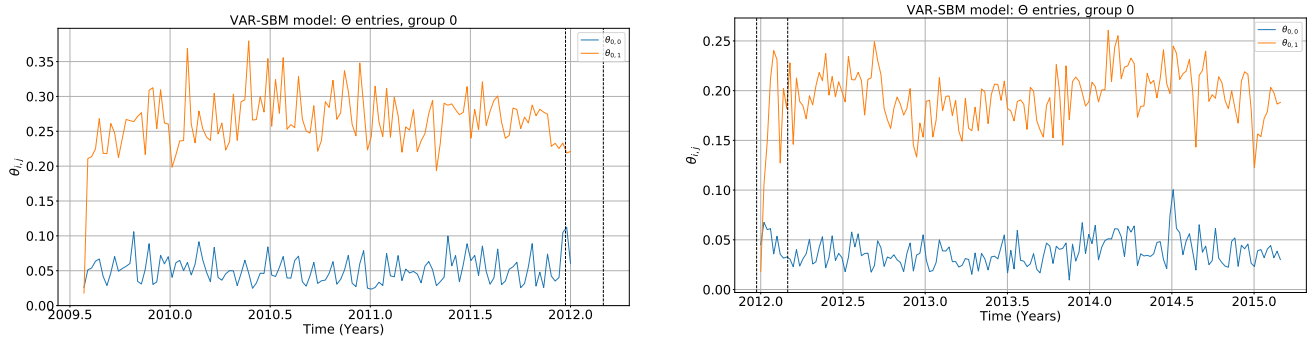

Figure 6.13: Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs

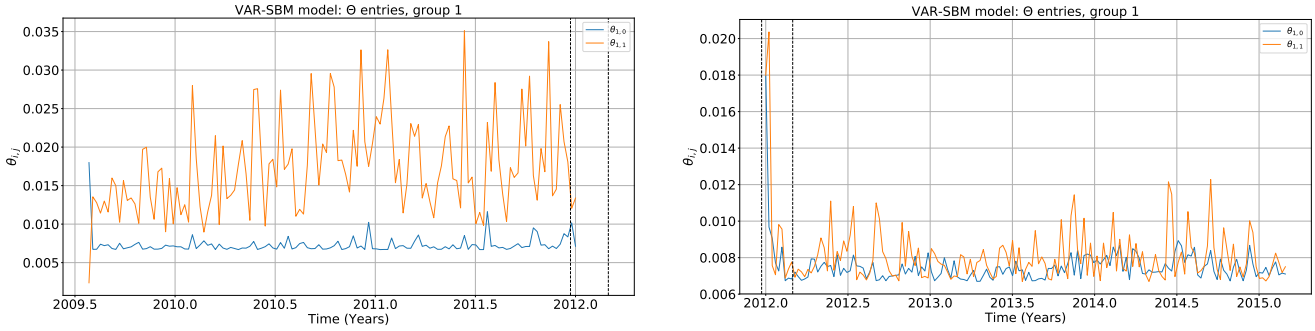

Figure 6.14: Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs

Evolution of the bank's behavior

In this Subsection, we analyze the evolution of the number of banks in each group. In the left panel of Fig. 6.15 we observe that the group of borrowers decreases in number during 2010 and the number of lenders and borrowers becomes is quite similar for a few weeks. We observe the same trend in the Xu and Hero model. In the period after the LTROs, the number of banks is very different and stationary, and the group of borrowers is bigger than the group of lenders.

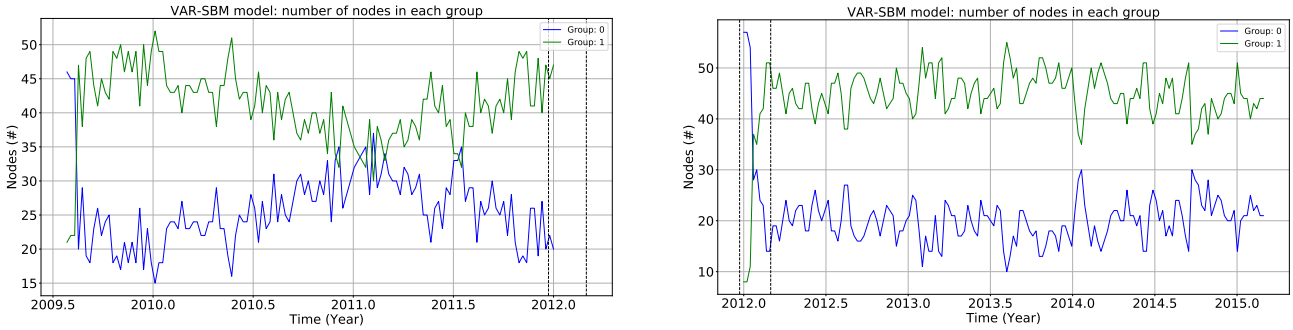

Figure 6.15: Evolution of the number of banks in the two groups. Blueline refers to the group of lenders and greenline refers to the group of borrowers. On the left panel, we see the period pre-LTROs, on the right panel the period post-LTROs

Results' validation

We employ the same metrics used for Xu and Hero model to validate the VAR-SBM model's inference. Since its evolution equation (Eq. 5.6) is different from the Xu and Hero one (Eq. 4.1, we adapt the computation of the predicted state in the algorithm 6.2.2 to Eq. 5.6. Hence, we obtain:

$$\hat{\boldsymbol{\psi}}^{t+1|t} = \hat{\boldsymbol{A}} + \hat{B} \cdot \hat{\boldsymbol{\psi}}^{t|t} \tag{6.4}$$

where $\hat{\boldsymbol{A}}$ is the inferred static vector of the process and $\hat{B}$ the inferred state evolution matrix.

MSE, Cosine Similarity, and Observed-Predicted Nodes

The analysis of the MSE is very similar to the previous one shown for the Xu and Hero model, where the metric obtain high scores. The following figures are all divided in two panel: one the left we analyze the first period, while on the right panel we observe the second period. In Fig. 6.16 we observe the observed number of links and the associated MSE at each time step. In Fig. 6.17 we observe the ratio between the MSE and the observed number of links. We observe that the VAR-SBM performs better in the first period respect to second period, where the ratio between MSE and the observed number of links reaches more time high percentages (17% at the 5^{th} time step, 19% at the 50^{th} time step, and 25% at the 150^{th} time steps). This results is in line with the Xu and Hero model's results, where we observe the same trend.

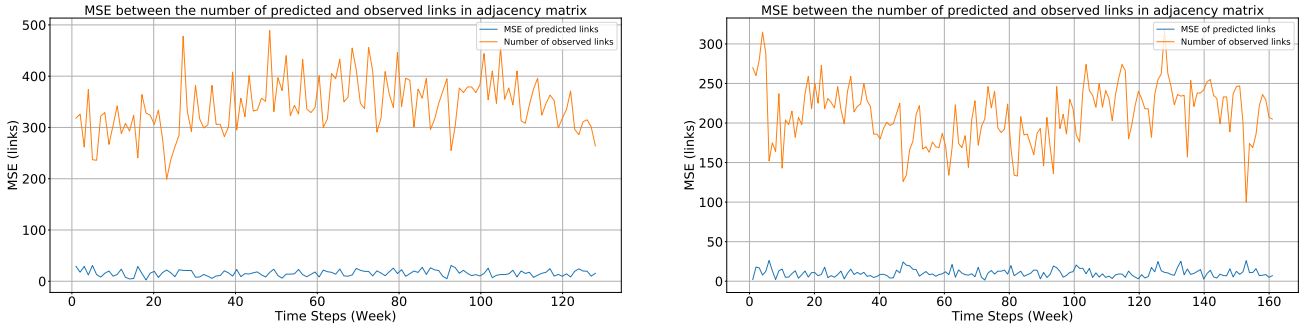

Figure 6.16: Analysis of the MSE: in orange the number of observed links in the network, in blue its MSE value at each time step

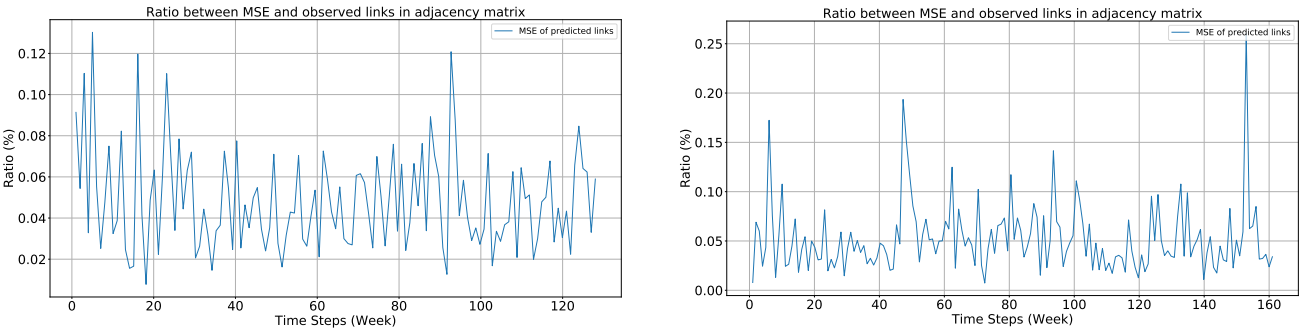

Figure 6.17: Analysis of the MSE: ratio between the blue and the orange lines in figure 6.16

In Fig. 6.18 we observe the cosine similarity for the VAR-SBM model. Also in this case we observe that the model performs better in first time window, where the minimum reached is 0.94, while for the second period we observe many values under 0.95.

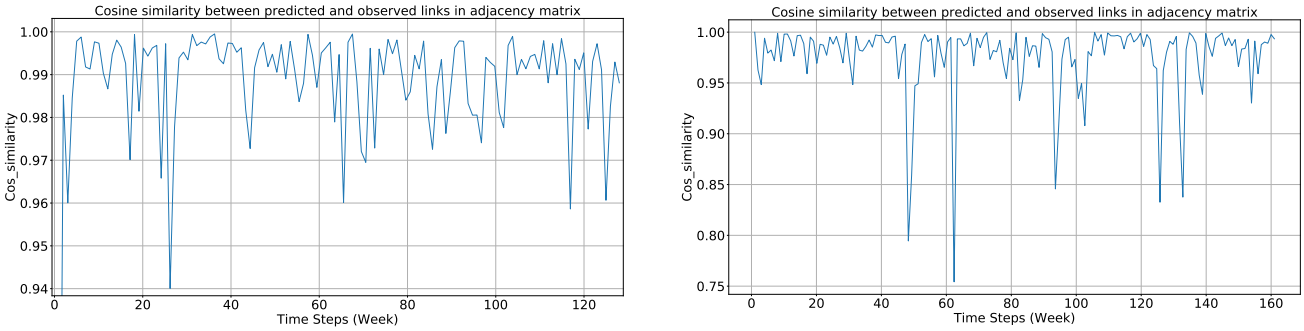

Figure 6.18: Cosine similarity between the predicted and the observed number of links for the VAR-SBM model

Concluding, we analyze the difference between the observed number of links and the predicted one for each entry of the affinity matrix Θ^t in Fig. 6.19 and 6.20. Unlike Xu and Hero model, we observe a large difference between the predicted values and the observed ones: this fact depends on the greater variability of the VAR-SBM model, since it depends on more hyperparameters respect to the other one.

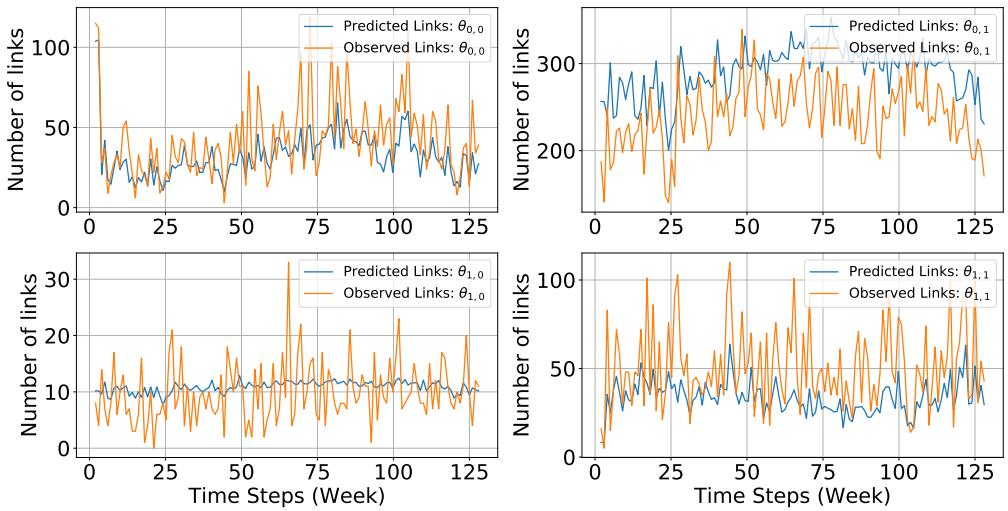

Figure 6.19: Number of observed and predicted links in VAR-SBM Model before the LTROs. The orange line refers to the observed links and blue line to the predicted links

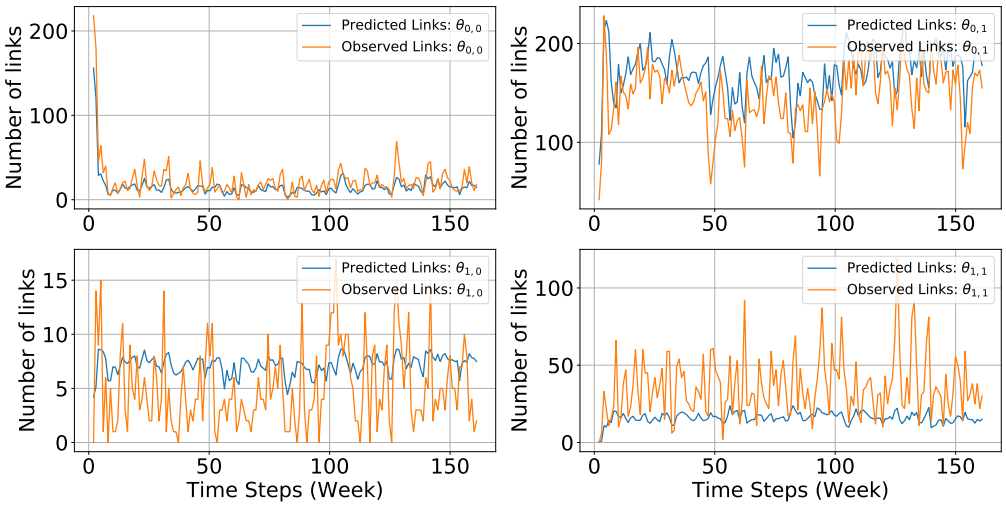

Figure 6.20: Number of observed and predicted links in VAR-SBM Model after the LTROs. The orange line refers to the observed links and blue line to the predicted links

6.3.3 Model comparison and conclusion

We summarize here the comparison of the two models employed on the e-MID network:

- Both models infer the same community structure, i.e. the bipartite structure, and the results are in line with Barucca and Lillo [5]. On the other hand, the affinity matrix Θ^t, inferred with Xu and Hero model, seems to describe better the dynamic of the network respect to the VAR-SBM model. This fact is supported by the analysis of the difference between the predicted and the observed number of links for each entry of Θ^t in the Subsections 6.3.1 and 6.3.2.

- The existence of an unexpected event (the LTROs) has strongly influenced the applicability of the VAR-SBM model: this fact has led to the split of the dataset into two distinct periods to analyze the periods like they describe a quasi-stationary process. This operation has two problems if applied in a generic case: firstly, in the case of an unexpected event, we have no idea if we have to split the dataset into many periods and where to do this choice, secondly, the creation of different time windows implies to generate many dynamic networks which describe the same dataset, and each of them has a lower number of time steps respect to the dynamic network generated with the complete dataset. As we studied in Section 5.4, the inference performance of the VAR-SBM model strongly depends on the number of time steps. On the contrary, the Xu and Hero model does not depend absolutely on the number of time steps and it could be used in the case of an unexpected event.

- The comparison of the analysis of the results' validation in section 6.3.1 and section 6.3.2 shows that the Xu and Hero perform better respect to the VAR-SBM model, especially in the analysis of the observed vs predicted number of links for each component of the affinity matrix. In fact, if we compare the results in Fig. 6.7 and 6.8 with Fig. 6.16 and 6.17, where we discuss the relation between the total observed number of links with its MSE (which is calculated respect to the total predicted number of links), we do not observe large difference between the two different models. On the contrary, the comparison of Fig. 6.9 respect to Fig. 6.19 and 6.20, we observe that difference between the predicted and the observed number of links for each entry of the affinity matrix in the Xu and Hero is lower respect to the VAR-SBM. Furthermore, the Xu and Hero model shows a more precise inference, by considering each time step.

Concluding, we observe that our results are consistent with the Barucca and Lillo's results in [5]. This means that three different algorithms (Xu and Hero model, VAR-SBM model, and MCMC) give the same results on the dataset using similar criteria of the study, strengthening the result in Barucca and Lillo [5]. Furthermore, the analysis of the structure of the network in Subsection 6.3.1 agree with the results shown in table 1 in [5]. Probably, a further analysis of the e-MID with different criteria (different number of groups, weighted network, undirected

network) could give more information on the structure of the e-MID network, but this is out of the scope of this thesis, and it is left for future research.

Chapter 7

Conclusion

Dynamic stochastic block models are useful to generate networks with very different structures and studying several networked systems, in many different research areas (physics, economy, sociology, math, and many others). Stochastic block models can be further generalized to the dynamic case by describing the time evolution of the state variables, then applying several tools of time series analysis.

In particular, the models we study (Xu and Hero model and the generalizations we have proposed in Chapter 5, namely AR-SBM model and VAR-SBM model) show are able to capture the main features of the networks, like the affinity matrix Θ and the labels of the nodes $\boldsymbol{c}$. The main differences between the Xu and Hero model and the other two models are the different networks' features they describe, like the dependency between the entries of the affinity matrix.

Comparing the models, we observe that:

- The Xu and Hero model could be applied to all dynamic networks independently form the length of the time series, since its model evolves according to Eq. 4.1, which relates the time steps only with the previous one. This represents one of the strengths of the Xu and Hero model. However, the Xu and Hero model displays some limitations: we must know a priori the evolution state matrix F^t and we also impose a structure to Γ^t to reduce the computational time. Furthermore, if the initial state is unknown or we can not make assumptions on it, the model must use an algorithm of the static stochastic blockmodel, which could be limited by the detectability problem (Chapter 3).
- The AR-SBM model and the VAR-SBM model require long time series of networks to infer the correct affinity matrices Θ^t and all the hyperparameters (Chapter 5). Thus, for a long time series, these models are more flexible to the Xu and Hero model. On the other hand, we must assume that the time series is stationary during its evolution.

As future outlooks, the AR-SBM model and the VAR-SBM model may be generalized to AR(p) and VAR(p) processes. These generalizations may infer more information respect to the

actual models.

Another area of study may be a detailed analysis of the sparse graphs. In fact, we studied the AR-SBM model and the VAR-SBM model with an affinity matrix with large probabilities of forming edges between nodes and with an assortative structure. Since we know that the performances of these models depend on the length of the time series, we may study which features should have the affinity matrix to be impossible to be inferred, independently of the length of the time series.

Appendix A

Computation of Kalman gain for non-diagonal matrix

The explicit result of the Kalman Gain K^t is not possible for a non-diagonal matrix Γ^t. This is due to the impossibility to invert explicitly a non-diagonal matrix of a two groups network, because it depends on 14 parameters (4 components of the vector $\boldsymbol{\psi}^{t|t-1}$, 4 component of the matrix $R^{t|t-1}$, 4 component of the matrix $\Sigma^{t|t-1}$ and the two different values s_{diag} and s_{nb} of the matrix Γ^t). The number of parameters depends on the squared number of the groups.

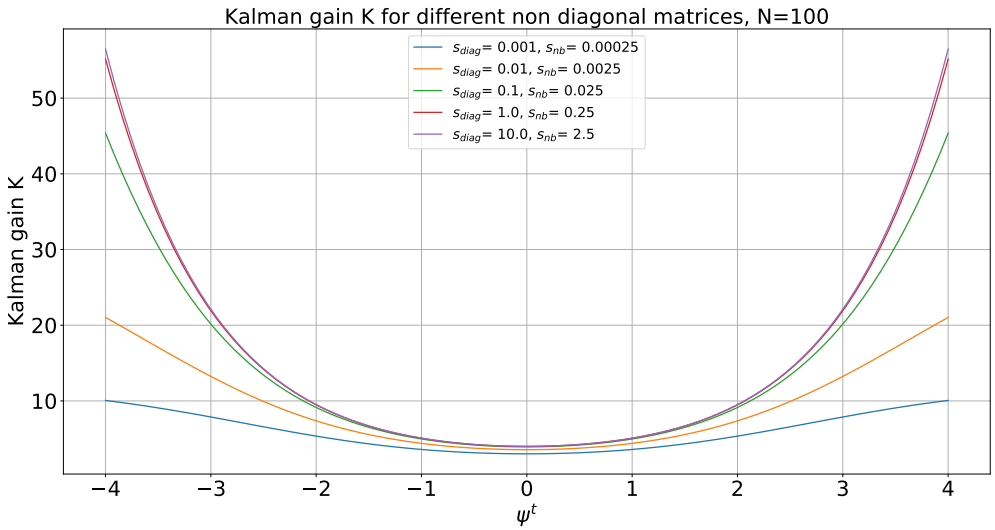

Figure A.1: Kalman Gain for $\langle N \rangle = 50$, $r = 0.004$

Anyway we simulate the Extended Kalman Filter with the non-diagonal Γ^t and we plot the results to show that our hypothesis in Chapter 5 is correct. We use the same parameters, so $R_{ii}^{1|0} = 0.004$. The network is divided into two groups and we progressively reduce the mean

number of nodes for each group.

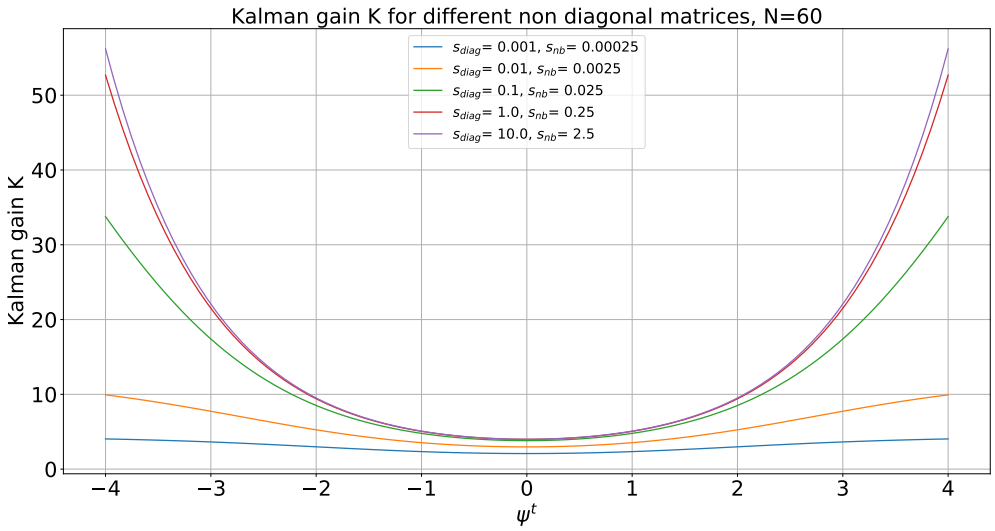

Figure A.2: Kalman Gain for $\langle N \rangle = 30$, $r = 0.004$

As we observe in Fig. A.1 and A.2, the Kalman Gain shows very similar values in the range $[-2, 2]$ as we have noticed in Chapters 5. Hence, for a non-diagonal matrix Γ^t, any value of s_{diag} and s_{nb} can be used, because the value of Kalman gain is similar for large part of the range of ψ, if the density of nodes for each group is high.

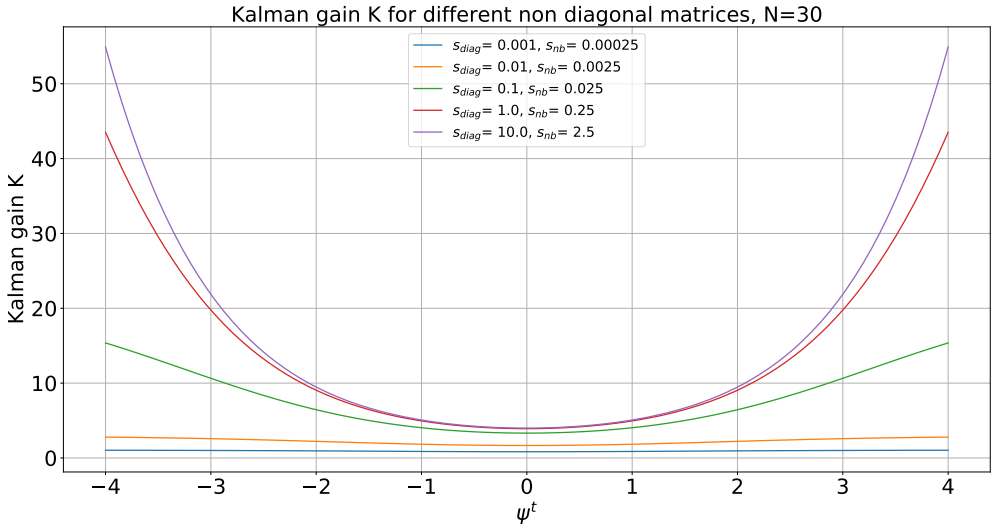

Figure A.3: Kalman Gain for $\langle N \rangle = 15$, $r = 0.004$

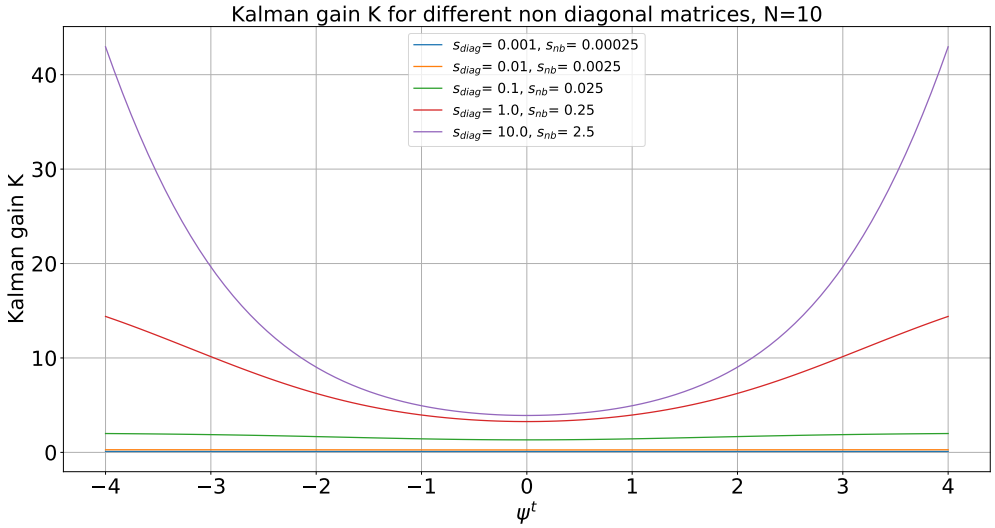

Figure A.4: Kalman Gain for $\langle N \rangle = 5$, $r = 0.004$

For the lower density of nodes for each group (Fig. A.3 A.4), the Kalman gain has different values for different Γ^t, as we observe in Chapter 5 for the diagonal Γ^t. This results supports our hypothesis of work in Chapter 5.

List of Figures

1.1 The Koningsberg bridges: on the left we observe the map of the city, on the right its graph representation 4

3.1 Example of undirected network 20
3.2 Example of directed network 21
3.3 Generative model and inference process 24
3.4 Erdos-Renyi network generated using the stochastic block model 24
3.5 Example of assortative (on the left) and dissortative (on the right) network, generated by the exchange of k_1 and k_2 25
3.6 Example of ordered (on the left) and Core-Periphery (on the right) networks . 26
3.7 Overlap for network with 2 and 4 groups varying the parameter ϵ^*[13] The graph compares the belief propagation results (red line) to MCMC (Monte Carlo Markov Chain) simulations (data point): on the left we observe data from simulations for large graphs, while on the right the graphs contain a small number of nodes 30

4.1 Logistic Function of ψ and its Taylor expansion 42
4.2 Evolution in time of ψ and θ, $s_{diag} = 0.001$ and $s_{snb} = 0.00025$ 43
4.3 Evolution in time of ψ and θ, $s_{diag} = 0.01$ and $s_{snb} = 0.0025$ 43
4.4 Evolution in time of ψ and θ, $s_{diag} = 0.1$ and $s_{snb} = 0.025$ 43
4.5 Evolution in time of ψ and θ, $s_{diag} = 1$ and $s_{snb} = 0.25$ 44
4.6 Kalman gain for $\langle N \rangle = 32$, $r = 0.004$ 46
4.7 Ratio between Kalman gains for different s_{diag}, $\langle N \rangle = 32$, $r = 0.004$ 46
4.8 Kalman gain for $\langle N \rangle = 100$, $r = 0.004$ 47
4.9 Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 100$, $r = 0.004$ 47
4.10 Kalman gain for $\langle N \rangle = 10$, $r = 0.004$ 48
4.11 Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 10$, $r = 0.004$ 48
4.12 Kalman gain for $\langle N \rangle = 5$, $r = 0.004$ 49
4.13 Ratio between Kalman gain for different s_{diag}, $\langle N \rangle = 5$, $r = 0.004$ 49
4.14 MSE for 2 groups and different Γ^t values. 50
4.15 MSE for 2 groups and different Γ^t values, focus on the range $20 < N < 50$ . . 51

4.16 ARI for 2 groups and different Γ^t values 52
4.17 ARI for 2 groups and different Γ^t values, focus on the range $20 < N < 50$ . . 52
4.18 MSPE for 2 groups and different Γ^t values 53
4.19 MSE for 128 nodes and different Γ^t values 54
4.20 ARI for 128 nodes and different Γ^t values 54
4.21 MSPE for 128 nodes and different Γ^t values 55
4.22 ARI evolution varying the percentage of leaving nodes 56
4.23 MSE evolution varying the percentage of leaving nodes 57

5.1 Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.001$ 65
5.2 Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.01$ 66
5.3 Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 0.1$ 66
5.4 Evolution of $\boldsymbol{\psi}^t$ and θ^t in time, for $s_{diag} = 1$ 67
5.5 Evolution of ARI for different numbers of nodes 68
5.6 Evolution of MSE of $\boldsymbol{\psi}^t$ for different numbers of nodes 68
5.7 Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes 69
5.8 Evolution of MSE of B for different numbers of nodes 70
5.9 Evolution of MSE of Γ for different numbers of nodes 70
5.10 Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.80$ and $s_{snb} = 0.05$ 72
5.11 Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.50$ and $s_{snb} = 0.05$ 72
5.12 Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0.20$ and $s_{snb} = 0.05$ 73
5.13 Evolution of $\boldsymbol{\psi}^t$ for 2 groups and 80 nodes in VAR-SBM Model, $s_{diag} = 0$ and $s_{snb} = 0.05$ 73
5.14 Evolution of ARI for different numbers of nodes in VAR-SBM model 74
5.15 Evolution of MSE of $\boldsymbol{\psi}^t$ for different numbers of nodes in VAR-SBM model . 75
5.16 Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes in VAR-SBM model . 76
5.17 Evolution of MSE of B for different numbers of nodes in VAR-SBM model . 76
5.18 Evolution of MSE of Γ for different numbers of nodes in VAR-SBM model . 77
5.19 Evolution of ARI for different numbers of nodes in AR-SBM model and VAR-SBM model 78
5.20 Evolution of MSE of $\boldsymbol{\psi}^t$ for different numbers of nodes in AR-SBM model and VAR-SBM model 78
5.21 Evolution of MSE of $\boldsymbol{A}$ for different numbers of nodes in AR-SBM model and VAR-SBM model 79
5.22 Evolution of MSE of B for different numbers of nodes in AR-SBM model and VAR-SBM model 80
5.23 Evolution of MSE of Γ for different numbers of nodes in AR-SBM model and VAR-SBM model 80

6.1 Weekly number of banks (left) and weekly traded volume in Million of Euros (right) traded ate-MID from June 2009 to December 2014. The blue line refers to all the banks while the green line refers only to Italian banks. Vertical red lines indicate the two LTRO measures by ECB on Dec. 22, 2011 and Feb. 29, 2012 [5] 85
6.2 Left. Weekly density (i.e. the number of links divided by the total number of possible links) of e-MID from June 2009 to December 2014. Right. Weekly volume per link (i.e. total volume in Million of Euros divided by the number of links. The blue line refers to all the banks while the green line refers only to Italian banks. Vertical red lines indicate the two LTRO measures by ECB on Dec. 22, 2011 and Feb. 29, 2012 [5] 86
6.3 Entries of the inferred affinity matrix Θ^t 89
6.4 Entries of the inferred affinity matrix Θ^t 90
6.5 Entries of the inferred affinity matrix Θ^t 91
6.6 Number of banks in each group: the green line refers to the borrowers and the blue line refers to the lenders 92
6.7 Analysis of the MSE: in orange the number of observed links in the network, in blue its MSE value at each time step. 93
6.8 Analysis of the MSE: the ratio between the blue and the orange lines in figure 6.7 93
6.9 Evolution of the observed and predicted groups of links for each component of $\boldsymbol{\Theta}$ 94
6.10 Cosine similarity for Xu and Hero model 94
6.11 Fraction of banks that leaves their groups at each time step in Xu and Hero model 95
6.12 Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs 96
6.13 Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs 97
6.14 Entries of the inferred affinity matrix Θ^t: in the left panel we observe the inferred parameters in the period pre-LTROs, in the right panel the inferred parameters after the LTROs 97
6.15 Evolution of the number of banks in the two groups. Blueline refers to the group of lenders and greenline refers to the group of borrowers. On the left panel, we see the period pre-LTROs, on the right panel the period post-LTROs 98
6.16 Analysis of the MSE: in orange the number of observed links in the network, in blue its MSE value at each time step 99
6.17 Analysis of the MSE: ratio between the blue and the orange lines in figure 6.16 99
6.18 Cosine similarity between the predicted and the observed number of links for the VAR-SBM model 100
6.19 Number of observed and predicted links in VAR-SBM Model before the LTROs. The orange line refers to the observed links and blue line to the predicted links 100

6.20 Number of observed and predicted links in VAR-SBM Model after the LTROs. The orange line refers to the observed links and blue line to the predicted links 101

A.1 Kalman Gain for $\langle N \rangle = 50, r = 0.004$ 106
A.2 Kalman Gain for $\langle N \rangle = 30, r = 0.004$ 107
A.3 Kalman Gain for $\langle N \rangle = 15, r = 0.004$ 107
A.4 Kalman Gain for $\langle N \rangle = 5, r = 0.004$ 108

Bibliography

[1] Emmanuel Abbe. "Community detection and stochastic block models: recent developments". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 6446–6531.

[2] Edoardo M Airoldi et al. "Mixed membership stochastic blockmodels". In: *Journal of machine learning research* 9.Sep (2008), pp. 1981–2014.

[3] Jeffrey Arnold et al. "The topology of interbank payment flows". In: *Federal Reserve Bank of New York, Staff Report* 243 (2006).

[4] Leonardo Bargigli et al. "The multiplex structure of interbank networks". In: *Quantitative Finance* 15.4 (2015), pp. 673–691.

[5] Paolo Barucca and Fabrizio Lillo. "The organization of the interbank network and how ECB unconventional measures affected the e-MID overnight market". In: *Computational Management Science* 15.1 (2018), pp. 33–53.

[6] Christopher Becher, Stephen Millard, and Kimmo Soramaki. "The network topology of CHAPS Sterling". In: (2008).

[7] Michael Boss et al. "Network topology of the interbank market". In: *Quantitative finance* 4.6 (2004), pp. 677–684.

[8] CERN. *Annual Report*. 2016. URL: `https://cds.cern.ch/collection/CERN%20Annual%20Reports`.

[9] CERN. *Annual Report*. 2017. URL: `https://cds.cern.ch/collection/CERN%20Annual%20Reports`.

[10] Aaron Clauset. *Network Analysis and Modeling, Lecture 16*. 2013.

[11] Aaron Clauset. *Network Analysis and Modeling, Lecture 17*. 2013.

[12] Rama Cont, Amal Moussa, et al. "Network structure and systemic risk in banking systems". In: *Edson Bastos e, Network Structure and Systemic Risk in Banking Systems (December 1, 2010)* (2010).

[13] Aurelien Decelle et al. "Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications". In: *Physical Review E* 84.6 (2011), p. 066106.

[14] Amir Dembo, Andrea Montanari, et al. "Ising models on locally tree-like graphs". In: *The Annals of Applied Probability* 20.2 (2010), pp. 565–592.

[15] James Durbin and Siem Jan Koopman. *Time series analysis by state space methods*. Oxford university press, 2012.

[16] John W Essam. "Graph theory and statistical physics". In: *Discrete Mathematics* 1.1 (1971), pp. 83–112.

[17] Ernesto Estrada. *Graph and Network Theory in Physics. A Short Introduction*. Tech. rep. 2013.

[18] Santo Fortunato. "Community detection in graphs". In: *Physics reports* 486.3-5 (2010), pp. 75–174.

[19] Daniel Fricke and Thomas Lux. "Core–periphery structure in the overnight money market: evidence from the e-mid trading platform". In: *Computational Economics* 45.3 (2015), pp. 359–395.

[20] Daniel Fricke, Karl Finger, Thomas Lux, et al. *On assortative and disassortative mixing in scale-free networks: The case of interbank credit networks*. Tech. rep. Kiel Working Paper, 2013.

[21] Simon Heimlicher, Marc Lelarge, and Laurent Massoulié. "Community detection in the labelled stochastic block model". In: *arXiv preprint arXiv:1209.2910* (2012).

[22] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. "Stochastic blockmodels: First steps". In: *Social networks* 5.2 (1983), pp. 109–137.

[23] Hajime Inaoka et al. "Fractal Network derived from banking transaction–An analysis of network structures formed by financial institutions". In: *Bank Jpn Work Pap* 4 (2004).

[24] Giulia Iori. *Interbank markets: network and rates*. 22nd Annual Workshop on Economic Science with Heterogeneous Interacting Agents Universita Cattolica. 2017.

[25] Giulia Iori, Burcu Kapar, and Jose Olmo. "Bank characteristics and the interbank money market: a distributional approach". In: *Studies in Nonlinear Dynamics & Econometrics* 19.3 (2015), pp. 249–283.

[26] Giulia Iori et al. "A network analysis of the Italian overnight money market". In: *Journal of Economic Dynamics and Control* 32.1 (2008), pp. 259–278.

[27] Aukosh Jagannath, Justin Ko, Subhabrata Sen, et al. "Max κ-cut and the inhomogeneous Potts spin glass". In: *The Annals of Applied Probability* 28.3 (2018), pp. 1536–1572.

[28] Rudolph Emil Kalman. "A new approach to linear filtering and prediction problems". In: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.

[29] Brian Karrer and Mark EJ Newman. "Stochastic blockmodels and community structure in networks". In: *Physical review E* 83.1 (2011), p. 016107.

[30] Elchanan Mossel, Joe Neeman, and Allan Sly. "Reconstruction and estimation in the planted partition model". In: *Probability Theory and Related Fields* 162.3-4 (2015), pp. 431–461.

[31] Mark Ed Newman. *Networks, An Introduction*. 2010. Chap. 1, Introduction.

[32] Krzysztof Nowicki and Tom A B Snijders. "Estimation and prediction for stochastic blockstructures". In: *Journal of the American statistical association* 96.455 (2001), pp. 1077–1087.

[33] William M Rand. "Objective criteria for the evaluation of clustering methods". In: *Journal of the American Statistical association* 66.336 (1971), pp. 846–850.

[34] Karl Rohe, Sourav Chatterjee, Bin Yu, et al. "Spectral clustering and the high-dimensional stochastic blockmodel". In: *The Annals of Statistics* 39.4 (2011), pp. 1878–1915.

[35] Kirsten Bonde Rørdam, Morten Linnemann Bech, et al. "The topology of Danish interbank money flows". In: (2008).

[36] R. Saporiti. *Quanti sono 4,8 zettabyte? Il traffico dati su Internet tra il 1984 ed il 2016*. 2018. URL: https://www.infodata.ilsole24ore.com/2018/12/13/internet-secondo-cisco-entro-2022-traffico-dati-triplichera/.

[37] Daniel L Sussman et al. "A consistent adjacency spectral embedding for stochastic blockmodel graphs". In: *Journal of the American Statistical Association* 107.499 (2012), pp. 1119–1128.

[38] Ruey S. Tsay. *Analysis of financial time series*. 2010.

[39] Yuchung J Wang and George Y Wong. "Stochastic blockmodels for directed graphs". In: *Journal of the American Statistical Association* 82.397 (1987), pp. 8–19.

[40] Kevin S Xu and Alfred O Hero. "Dynamic stochastic blockmodels for time-evolving social networks". In: *IEEE Journal of Selected Topics in Signal Processing* 8.4 (2014), pp. 552–562.

[41] Yunpeng Zhao, Elizaveta Levina, Ji Zhu, et al. "Consistency of community detection in networks under degree-corrected stochastic block models". In: *The Annals of Statistics* 40.4 (2012), pp. 2266–2292.