Complex control systems: an information theoretic approach

Toupance, Pierre-Alain Daniel T.

Complex control systems: an information theoretic approach

Pierre-Alain TOUPANCE
France
Thèse - 5857 -

Thesis submitted in partial fulfillment of the requirements for the degree of
Docteur en sciences de l'ingénieur
and the degree of
Docteur de la communauté Grenoble Alpes
with the speciality *EEATS – Automatique Productique*

Docteur ès science, mention informatique
Docteur de l'université de Genève
Centre Universitaire d'Informatique (CUI)

Dissertation committee:

Pr. Franco Bagnoli	Università di Firenze	(**Examiner**)
Pr. Timoteo Carletti	Université de Namur	(**Reviewer**)
Pr. Jean-Charles Delvenne	Université Catholique de Louvain	(**Reviewer**)
Pr. Samira El Yacoubi	Université de Perpignan Via Domitia	(**President**)
Dr. Jean-Luc Falcone	Université de Genève	(**Examiner**)
Dr. Paolo Frasca	CNRS (Gipsa-lab)	(**Examiner**)
Pr. Bastien Chopard	Université de Genève	(**Co-supervisor**)
Pr. Laurent Lefèvre	Grenoble INP, Univ. Grenoble Alpes	(**Co-supervisor**)

Grenoble, august 2024.

UNIVERSITÉ
DE GENÈVE
FACULTÉ DES SCIENCES

DOCTORAT ÈS SCIENCES, MENTION INFORMATIQUE

En cotutelle avec
L'UNIVERSITÉ GRENOBLE ALPES, SAINT-MARTIN-D'HÈRES, FRANCE

Thèse de Monsieur Pierre-Alain Daniel T. TOUPANCE

intitulée :

**«Commandabilité des systèmes complexes :
une approche par la théorie de l' information »**

La Faculté des sciences, sur le préavis de

Monsieur B. CHOPARD, professeur ordinaire et directeur de thèse
Département d'informatique

Monsieur L. LEFEVRE, professeur et codirecteur de thèse
École nationale supérieure en systèmes avancés et réseaux - Grenoble INP-UGA,
Valence, France

Monsieur J.-L. FALCONE, docteur
Département d'informatique

Monsieur T. CARLETTI, professeur
Département de mathématique, Faculté des sciences, Université de Namur,
Namur, Belgique

Monsieur J.-C. DELVENNE, professeur
Institute of Information and Communication Technologies, Electronics and
Applied Mathematics (ICTM), Université catholique de Louvain, Louvain-la-Neuve, Belgique

Madame S. EL YACOUBI, professeure
Department of Mathematics and Computer Science, University of Perpignan,
Perpignan, France

Monsieur F. BAGNOLI, professeur
Dipartimento di Fisica e Astronomia, Università degli Studi di Firenze, Firenze, Italia

Monsieur P. FRASCA, docteur
GIPSA-lab, Centre national de la recherche scientifique (CNRS), Institut d'ingénierie et de management - Grenoble INP - UGA, Université Grenoble Alpes, Saint Martin d'Hères, France

autorise l'impression de la présente thèse, sans exprimer d'opinion sur les propositions qui y sont énoncées.

Genève, le 7 novembre 2024

Thèse - 5857 -

La Doyenne

N.B. - La thèse doit porter la déclaration précédente et remplir les conditions énumérées dans les "Informations relatives aux thèses de doctorat à l'Université de Genève".

Résumé

Nous nous sommes intéressés aux problèmes de la théorie du contrôle de systèmes complexes. Nous avons abordé le lien entre la commandabilité ou l'observabilité d'un système dynamique complexe et les concepts de la théorie de l'information. Le modèle choisi pour les applications numériques est un modèle dynamique de vote défini sur un graphe "scale free" qui représente les liens sociaux entre agents.
La causalité temporelle entre les comportements (états) des agents dans ces systèmes a pu être quantifiée par la notion d'*information mutuelle décalée* dans le temps. Nous avons également défini la *multi-information décalée* dans le temps qui permet de quantifier l'influence d'un agent sur l'ensemble des agents du système.
Cela nous a offert un moyen d'orienter les actions pour faire tendre le système vers un état souhaité. Nous avons également souligné l'importance du bruit lors de la commande du système, et montré que le rayon d'action d'un agent, caractérisé par une longueur de contrôle, était limité. Nous avons fait le lien entre cette limite et la notion de capacité d'un canal de transmission en théorie de l'information.
Ensuite, nous avons montré que l'information mutuelle décalée dans le temps nous permettait de retrouver la topologie du graphe des interactions au sein du système à partir de la mesure du comportement individuel des agents au cours du temps. Nous nous sommes également intéressés à la détection de changement de topologie du système, en calculant l'information décalée dans le temps sur une fenêtre temporelle glissante.
La théorie de l'information nous a permis également de déterminer une décomposition du système en communautés définies sur la base de l'information mutuelle entre agents. La partition du système obtenue a permis d'envisager une réduction du système par agrégation de ces communautés. Cette réduction a été testée sur des automates cellulaires 1D, puis généralisée au cas d'un automate probabiliste défini sur un graphe quelconque. Nous avons utilisé l'information mutuelle pour qualifier la qualité de cette réduction.

Mots clefs

Systèmes dynamiques complexes, théorie de l'information, modèle probabiliste, théorie du contrôle , graphes, communautés, modèle de vote, automates cellulaires, réduction.

Abstract

In this thesis, we focused on the problems of control theory in the context of complex systems. We addressed the link between the controllability or observability of a complex system and information-theoretic concepts. The model chosen for the numerical applications is a voter model based on the representation of social links (influence) in a scale-free graph. We first looked at the temporal causality between agents in these systems, which could be quantified by the *delayed mutual information* defined in this thesis. We defined the *delayed multi-information*, which quantifies the influence of an agent on the whole system. This approach provided us with a mean of directing actions to move the system towards a desired state. We have also highlighted the importance of noise in this control process, and shown that an agent's radius of action is limited and characterized by a *control distance*. We linked this control limit to the notion of transmission *channel capacity* as it is defined in information theory.
The delayed mutual information also allowed us, without a priori knowledge on the system topology, to obtain structural information about the network: we were able to recover the topology of the interconnection graph from the evolution of the agents' states. This approach allows to detect changes in system topology over time, by calculating the delayed mutual information over a sliding time window. Finally, information theory enabled us to determine a decomposition of the system into communities. The resulting partition of the system into communities suggests a reduction by agregation of the system. The principle of this reduction was first presented and tested on 1D cellular automata (CA), and then generalized to probablistic CA defined on a graph. We used mutual information itself to assess the quality of the reduction.

keywords

Complex dynamical systems, information theory, probabilistic models, control theory, Mutual and multi-information, graph, communities, voter model, Cellular Automata, coarse-graining

List of publications

Peer-reviewed journals

- P.A. Toupance, B. Chopard, L. Lefèvre , Influence measurement in a complex dynamical model : an information theoretic approach, *Journal of Computational Science, 44:101115 2020*

- P.A. Toupance, B. Chopard, L. Lefèvre, System reduction and controlability: an approach based on cellular automata, *Natural Computing, pages 1 - 13, 2023*

Proceedings of international conferences

- P.A. Toupance, B. Chopard, L. Lefèvre, Identification of complex network topologies through delayed mutual information, *IFACPapersOnLine, 53(2):1019–1024, 2020.* [106]

- P.A. Toupance, B. Chopard, L. Lefèvre, Detection of topology changes in dynamical system : an information theoretic approach, *In Proceedings of the conference ACRI 2020, Lodz, Poland, Poland, December 2–4, 2020, Proceedings 14, pages 26–35. Springer, 2021.*

- P.A. Toupance, B. Chopard, L. Lefèvre, System reduction and controlability: an approach based on cellular automata, *In Cellular Automata: 15th International Conference on Cellular Automata for Research and Industry, ACRI 2022, Geneva, Switzerland, September 12–15, 2022, Proceedings, pages 94–105. Springer, 2022.*

Remerciements

Je tiens à exprimer mes remerciements en premier à mes directeurs de thèse Bastien CHOPARD et Laurent LEFEVRE. Je vous remercie tout d'abord de m'avoir proposé de vivre cette belle aventure. J'ai apprécié votre disponibilté, votre patience, vos conseils, votre compréhension dans les moments où j'ai eu le plus d'incertitudes, ou quand les aléas de la vie ont freiné mes avancées. Les échanges avec vous qui ne se limitaient pas au sujet de la thèse ont été très enrichissants pour moi, un grand merci pour tout cela. Je garderai de très bons souvenirs de tous ces moments passés ensemble. Laurent, je te remercie pour la double casquette de directeur de thèse et d'ami que tu as su gérer parfaitement. Bastien, je te remercie pour tous tes conseils éclairés, ton accueil lors de mes venus à Genève, et ta bienveillance.

Je voudrais remercier Jean-Charles DELVENNE et Timétéo CARLETTI d'avoir accepté d'être rapporteur, leurs remarques constructives m'ont éclairé sur de nombreux points et m'ont permis, je l'espère, d'améliorer le contenu de mon manuscrit.

Je tiens à remercier Mme Samira EL YACOUBI de l'université de Perpignan pour l'honneur qu'elle m'a fait en présidant mon jury de Thèse, ainsi que Pr. Franco BAGNOLI de l'Université de FLORENCE, Dr. Jean-Luc FALCONE de l'université de GENEVE et Dr. Paolo FRASCA du CNRS d'avoir participer à mon jury.
Je te tiens à remercier Franck pour tous ses conseils précieux lors de la préparation de la soutenance.

Je souhaite également remercier l'ESISAR et Grenoble INP qui m'ont permis de mener à bien ce projet, en me libérant du temps et en tenant compte de toutes les contraintes que cette thèse engendraient dans ma vie professionnelle.
D'autre part, j'ai grandement apprécié l'accompagnement de l'école doctorale durant ces années, je remercie particulièrement Jumana BOUSSEY et Sylvie CHARBONNIER pour leur bienveillance et leur suivi.
Je remercie tous mes amis pour leur présence et tous les bons moments partagés, même si, durant cette période je n'ai pas été beaucoup disponible.

Benoît, cela fait plus de 45 ans que l'on se connait, je mesure chaque jour la chance de t'avoir comme ami depuis tant d'années.

Nicolas, Vincent et Arnaud, nous avons commencé nos études ensemble, avec cette thèse, je les ai terminées bien plus tard que vous, je vous remercie d'avoir toujours été présents et de m'avoir accompagnés durant toutes ces années.
Alexandra et Alain, je tiens à vous dire que nos agapes lyonnaises sont toujours de très bons moments, même si durant ces dernières années, ces moments furent plus rares.

Marc, je te remercie pour ton amitié, ta présence et le partage de ton expérience qui ont été pour moi d'une grande aide. Le repas du 10 mars 2020 restera pour moi un souvenir fort.

Franck, je te remercie pour ton amitié forte et sincère, et pour tout ce que ça engendre.

Les copines de l'ALE, je vous remercie pour tous ces bons moments en meute, à pied, en vélo, en ski ou autour d'une bière.

Je tiens à remercier ma famille, mes parents pour tout ce que vous m'avez apporté pour votre soutien et pour les valeurs que vous avez essayées de me transmettre. Ma soeur, un grand merci pour ton soutien et tout ce que tu m'apportes, j'ai grandement apprécié ta présence le jour de ma soutenance, c'était important pour moi.

Mes filles, je vous remercie pour tous les bons moments que vous m'avez offerts, j'en profite pour vous dire que je suis fier de vous. Hannah et Raphaëlle, vous me manquez. Adèle, je me doute que ça n'a pas dû être toujours simple pour toi, les rôles ont parfois été inversés durant ces dernières années, beaucoup de choses reposaient sur toi, sache que ta bonne humeur, ton enthousiasme, ta vision de la vie toujours positive me rendent heureux.

Agnès, j'ai conscience que ce n'était pas simple de supporter mes moments de découragement, mes doutes et tout ce qui en résultait. Je te remercie pour ta présence, ta patience, ta clairvoyance. Les mots que tu as su choisir pour m'encourager durant ces dernières années ont eu un rôle très important dans ce projet. Pour tout cela, et plein d'autres choses, merci.

Contents

List of Figures

Notations

Symbol	Description
$\mathbb{N}$	set of natural integers
$\mathbb{Z}$	set of integers
$\mathbb{R}$	set of real numbers
$\mathbb{C}$	set of complex numbers
$\lfloor x \rfloor$	Lower integer part of x
$[\![p; q]\!]$	set of integers $\{p, p+1, ..., q\}$
$M_{p,q}(\mathbb{R})$ or $\mathbb{R}^{p\times q}$	Set of real matrices with p rows and q columns
$M_n(\mathbb{R})$ or $\mathbb{R}^{n\times n}$	Set of real matrices with n rows and n columns
A^T	the transpose of the matrix A.
$[A_1; A_2; \dots; A_p]$	The matrix obtained by concatenating the matrices $A_1,\ A_2, ..., A_p$
$0_{n,p}$	The null matrix with n rows and p columns.
$x_{1:n}$	The vector $(x_1, x_2, ...x_n)$
$G = (V, E)$	G is a graph, V the set of vertices, E the set of edges.
p_x	$\mathbb{P}(X = x)$.
p_{xy}	$\mathbb{P}((X, Y) = (x, y))$.
$P_A(B)$ or $P(A\|B)$	The probability of event A knowing B
K_n	A complete graph of order n.
C_n	A circular graph of order n.
$d(x)$	degree of the vertex x.
$\delta(G)$	diameter of the graph G.
$H(X)$	Entropy of the random variable X.
$H(X_1, X_2, ..., X_n)$	Mutual entropy of the vector of random variables $X = (X_1, ...X_n)$.
$H_2(\epsilon)$	Binary entropy with parameter epsilon, i.e. $-\epsilon \log \epsilon - (1-\epsilon)\log(1-\epsilon)$.
$H(X\|Y)$	Conditional Entropy of the random variable X knowing Y.
$I(X, Y)$	Mutuel information of the random variables X and Y.
$w_{i,j}(t, \tau)$	τ-delayed Mutual information of agents i and j at the moment t.
$w_i(t, \tau)$	τ-delayed Mutual multi-information of the agent i at the moment t.
$L^2(I, J)$	set of functions $f : I \to J$ such that $\|\|f\|\|^2$ is integrable on I.
$\Sigma = (A, B, C)$	system defined by the functions or matrices A, B and C .
$\phi(u, x_0, t)$	state of the system at the moment t with the initial state x_0.
W^c	Grammian Controllability.
W^o	Grammian Observability.

1 | Introduction

The aim of this thesis is to show that information theory offers tools for determining solutions to certain problems in control theory. Claude Shannon, the founder of information theory, used probabilities to quantify information in order to provide tools in the field of telecommunications. The first applications of this probabilistic modelling were in information coding, data compression and cryptography [93], [64], [46], [92]. Information theory also makes it possible to quantify the causal relationships between data. Determining the actions that enable a system to be controlled requires a good knowledge of these causal links. The knowledge of both the network topology and the laws governing the dynamics are useful for the control of complex systems. As we will show, information theory offers the means to take these two factors into account.

Control theory is a well-established field, based on several mathematically well-defined concepts. Controllability and observability are introduced by Kalmann [48] [47] in the case of a finite-dimensional linear system. Controllability and its dual form, observability, are major subjects of study in control theory. From a practical point of view, these concepts, when it is not easy to define it formally due to the complexity of the dynamics, can be defined through numerical and optimization procedures. For example, the cost of control, usually expressed with Grammians, is related to the energy required to control a system and can be computed by numerical methods for discrete non-linear systems, based on their physical interpretation (see, for example, the economic model described in [89]). In this thesis, we were interested in investigating issues related to the controllability and observability of complex systems. Their large size and their non-linear dynamics make them challenging to analyze. We made use of a simple voter model throughout the thesis to get empirical results through simulations and to give us some intuition for this analysis.

When dealing with the observability/controllability of complex systems, at least two approaches have been considered. In the first one, dealing with large unstructured (although often sparse) dynamical systems, numerical algebra techniques have been developed either for the reduction of the systems or for the development of suitable observability/controllability analysis tools [4], [9]. In this approach, only numerical results are obtained and they may not be easily related to the structural properties of the considered complex system (a.o. the interaction topology). The second approach has been initiated by Siljak [99] and work has been continued by Lin and his co-authors [56]. In this approach, structural dynamical properties, including observability and controllability, are defined as properties which are satisfied for almost all numerical

values of the parameters in the model, that is properties which are essentially depending on the system interconnection topology. A graph describing the structure of the interaction between the nodes of the system (sometimes referred to as the inference diagram) is then defined and control properties are related to graph theoretic properties which may be decided using existing efficient graph algorithms (see [100] for an introductive textbook). Recent surveys of the results obtained with this approach are [58], [61] or [59], for specific aspects related to large complex networks.

Although suitable for complex systems and providing results on structural dynamical properties, the graph theoretic approach exhibits several limitations which still need to be overcome. Three of them are related respectively to the existence (in many application examples) of constraints on the control perturbations and variations in the system interconnection topology.

In this thesis, to study complex systems, we focused on the causal links in the network. The main difficulty encountered when trying to use the standard tools of probability theory to quantify the causal influence of one entity on another is due to "confusion". Suppose that A has a direct influence on B and C, causing these agents to act in a correlated manner. This correlation is easy to detect, but when it comes to interpreting it, it can be attributed either to B's direct influence on C, or to the existence of a common ancestor influencing both (this situation, which is the true one in our example, is called *confusion*). Of course, direct influence and confusion can coexist in even moderately complicated systems, making the situation even more difficult to disentangle. While some authors have proposed to develop a "causal calculus" that takes causal ordering into account [40], more energy has been devoted to capturing causal relationships using "classical" tools by defining refined versions of correlation measures [96]. In particular, many of these proposals use quantities introduced mainly in information theory, such as "mutual information" and "transfer entropy" [96].

Information theory thus offers tools for quantifying causal relationships between data. By observing the system behavior over time, we will show that the delayed mutual information between a variable and the rest of the system is strongly correlated with the ability of that variable to control the system, at least for the well-known voter model with-noise. Importantly, this measure of information can be applied without any knowledge of the rules governing system dynamics or interaction topology, "simply" observing the agents' state. This study suggests that the degrees of freedom of a complex dynamical system can be classified according to their level of influence, measured from delayed multi information. The level of influence of a given state variable can be used to determine the set of most important variables governing the temporal evolution of a system. In particular, we will show that the behavior of the most influential variables provides an early warning signal in the voter model. Previously existing results give negative answers about the possibility to control a system only based on

the structure of the underlying interactions, without considering constraints on cost of the control and the role of noise. We have therefore analyzed the impact of noise on the influence of a given state variable and showed that the delayed multi-information decreases with noise similarly to the reachability Gramians (a usual control measure) until practical controllability is lost.

We were also interested in complex systems whose topology evolves over time. This is a problem that appears in many practical situations where some links are created, while others disappear. For example, in market systems, the topology naturally varies according to the evolution of interactions between social (e.g. economic) agents. In complex control systems, the temporal aspect often stems from technological choices, for example when dynamic routing protocols are used (ad hoc wireless sensor/actuator networks) or because reconfigurations based on discrete events are required (fault-tolerant control). Structural controllability and observability are clearly affected by these changes in the interconnection topology. The problem of inferring the topological structure of systems from their measured outputs has already been studied in [42], [1] [102], [89], mostly for small dynamic Boolean networks. In this thesis, we propose to use delayed mutual information computed on sliding time window in order to detect topology changes for large systems more general than boolean networks. In order to study a way of controlling a system, we're interested in a classic problem: determining a community partition of the graph representing the network. Our approach is a little different from conventional methods, in that the elements of the partition we construct are linked to the dynamics of the system and do not depend solely on the topology. We will define these influence community sets to differentiate them from the classical notion of communities.

Finally, we also considered the problem of coarse-graining complex systems. The goal was to explore the possibility to simplify the dynamics, by reducing the number of degrees of freedom. We had in mind the possibility to consider the control of a reduced system as a good approximation of the control of the full system. For this purpose, the most popular reduction methods in the literature are generally based on matrix decomposition algorithms and projections that preserve certain properties of the system (see for example [4]). We rather focused on *reduction by aggregation* methods more specifically adapted to complex systems where information is available only on the system topology and local dynamics. We first considered, as an illustration example, a voter model defined on a periodic one-dimensional topology (with only nearest neighbors interactions), then generalized the investigations to systems defined on graphs. We proposed a reduction method based on the partitioning of the graph into communities, defined with respect to delayed mutual information. We then analyzed whether the coarse-grained model is a good approximation of the original one and

what good approximation means in this context.

Organization of the manuscrit

The manuscript is organized in 6 Chapters including introduction (chapter 1) and conclusion (chapter 6).

Chapter 1 presents the context of the thesis. Chapter 2 introduces some basic concepts necessary for a good understanding of the thesis. Personal contributions are presented in chapters 3, 4 and 5. Finally, chapter 6 presents the conclusion of the work and the future prospects.

Chapter 2 lays the foundations for understanding the other chapters. We begin by presenting complex systems and their representation by graphs. In the first section, we give some basic notions of graphs and discuss the tools that allow graphs to be studied structurally. In the following section, we define some principles of control theory: controllability, reachability and observability. We give the Kalman criteria. We also discuss the problems associated with large systems, which require an in-depth study of the structure of these networks in order to control them, and sometimes to simplify them. To conclude this chapter, we present elements of Claude Shannon's information theory, which we will use in the following chapters to propose ways of controlling complex systems.

In chapter 3, we make the link between controllability and information theory. After addressing the notion of causality, we define delayed mutual information, which enables us to quantify the temporal causality between two events, and present the means of calculating it by sampling : we observe the behavior of the system to obtain our samples, and then we use sampling method to obtain the probability which allows us to calculate the mutual information.[1] We then define the voter model, which will be used as the running example throughout the thesis. The chapter ends with the first results obtained on this voter model, related to the influence of the agents (nodes) and its relation with control.

In chapter 4, we focus on system topology. We show that by observing the behavior of the system, and by calculating the delayed mutual information and multi-information, we are able to recover the overall structure of the system. We also show that it is possible to detect topology changes over time. Finally, we see that information theory allows us to obtain an original way of decomposing a graph representing a system into communities of influence.

In chapter 5, we present a method for simplifying a system by coarse graining. We first

[1] We also define multi-information, which allows us to quantify the influence of an agent on the whole dynamical complex system. As we shall see in the definition of a complex system (def. 2.1.1, an agent is the name often given to a component of the system.

apply this simplification to deterministic cellular automata, and then apply it to a probabilistic system: the 1D voter model. Using mutual information, we quantify the error induced by the reduction of the system. We conclude the chapter with the application of the proposed coarse-graining method to the the example of a voter model defined on a scale-free graph, using the community of influence decomposition obtained in the previous chapter.

2 | Basic concepts

2.1 Dynamical complex systems on graphs

2.1.1 Complex systems

Definition

Complex systems are used as models in many disciplines: biology, physics, chemistry, human sciences (linguistics, economics, sociology, psychology, etc.) or computer science. We can give as classical examples of complex systems the economy and financial markets, the brain, road traffic, the Internet and ecosystems. It is difficult to give a precise technical definition of what complex systems are. We will therefore use the following definition, which covers the main characteristics of complex systems as outlined by Ladyman and Lambert in their paper "What is a complex system?" ([53]).

Definition 2.1.1. Complex systems are composed of a large number of components, often named agents, that can be heterogeneous. There are many interactions between the components and these interactions are non-linear and lead the system to have behaviors that cannot be trivially inferred from the behavior of an isolated agent.

According to Newman [76], complex systems theory is divised into two approaches, each of them with specific methodology:

- the creation and study of simplified models can use of dynamical systems theory, game theory, information theory, cellular automata, networks, computational complexity theory, and numerical methods. These approximate models do not provide the exact behavior of the system, but rather focus on the most important data (reducing the computation time required for simulations).
- the creation of more comprehensive and realistic models. This approach use quite often uses Monte Carlo method or agent-based simulation methods.

The idea of the complexity of a system may be linked to different characteristics such as non-linear interactions, multiple interactions between many different components, a system or component that by design or function or both is difficult to understand and verify, the size of the system. These characteristics have for consequence the unpredictability of the behavior of the system. This unpredictability is related to the difficulty of calculating states over time, either because of the large size of the system

or because of the dynamics. In the paper "What is a complex system" [53], Ladyman and Lambert propose to quantify complexity using entropy, which is an information theoretic metric that quantifies unpredictability. On the other hand, complex systems have a sophisticated internal causal architecture that stores and processes information. Information theory seems thus seems a natural approach to study these complex system [32]. Some concepts from information theory will be presented in section 2.3. To model the causal structure in complex systems, we will also use another field of mathematics: graph theory [73].

Graph theory to analyze complex systems

Graph theory offers methods to describe the structure and analyze complex systems. Graphs are defined by nodes and links between them. Links can represent physical relationships between nodes or represent causal links between node states.
For example in [103], Spulber represents the brain by a graph where the nodes are the neuronal elements (neurons) that are linked by edges representing physical connections (synapses). Becq et al present a method for the estimation of oriented graphs of connectivity between signals from different areas of the brain [3].
In the field of economics, graphs are also used to represent complex systems that allow us to model economic relationships. For example, in the 1970s, as part of the Club of Rome's research program, an attempt to describe the world system was proposed, called the world3 model [67], [90] . This model simulates the interactions between world population, industrial growth and food production within the limits of the earth ecosystem. There are 315 variables divided into 5 subsystems: population, capital, pollution, agriculture, non renewable resources. The evolution of these parameters is characterized by differential equations. Figure 2.1 shows us a graph representing the interactions between the sub-systems, and indicates the exogenous variables in each sub-system that originate from the other sub-systems. For example, we can see in this figure that the variable IO (Industrial Output), which is the quantity of production of material, variable from the sub-system capital, influences the values of certain variables in the agriculture sub-system, and that the world population (POP) has an influence on variables in all the sub-systems. The list of variables and the equations that define the dynamics of the world3 model are detailed in the paper [30].
From the simulations of this model, Meadows et al proposed a report on the limits of growth ([66]).

The last example we will present is the epidemic propagation model. It can be represented by a graph where each node is a person and the physical links between people that allow contamination of a healthy person by an infected person are represented by the edges. In the case of the SIR model, there are 3 possible states for each vertex: S

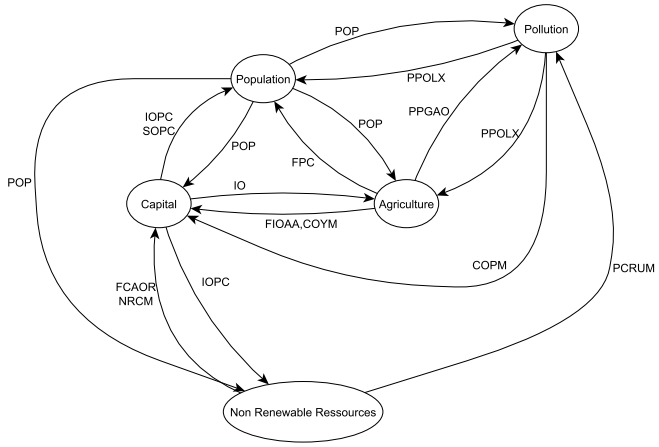

Figure 2.1 – Interaction between world3 model subsystems: the valuations of incoming arcs are the exogenous variables of the subsystems that influence them.

(Susceptible) , I (Infected) or R (Recovered).
Knowing the epidemic's propagation dynamics, the model can be used to simulate its spread, testing scenarios (containment, detection, vaccination) in an attempt to limit its spread.

We have just presented three examples, but the fields of application are very diverse (automobile traffic [95], [8], biology [50], economics [19], [35], ...).
In the next section, we define the concept of graphs and present graph-theoretic tools used to study the structure of complex systems. This structural analysis enables us to predict and understand system behavior. It is also important when it comes to controlling the system.

2.1.2 Dynamical systems on graphs

We will start this section by defining undirected graphs, followed by directed graphs:

Definition 2.1.2. An undirected graph is a pair $G = (V, E)$ with V is a non empty finite set of elements called vertices and E is a set of pairs of elements of V, called edges.
Let $V = \{x_1, x_2, ..., x_n\}$ and $E = \{\{x, y\}, (x, y) \in E^2$ and $\{x, y\}$ is an edge$\}$
x and y are neighbors or adjacent if $\{x, y\}$ is a edge of G.
We note:

- N_x the set of neighbors of x i.e. $N_x = \{y \in V, \{x, y\} \in E\}$.
- E_x the set of edges at x i.e. $E_x = \{\{x, y\}, y \in V \text{ and } \{x, y\} \in E\}$.

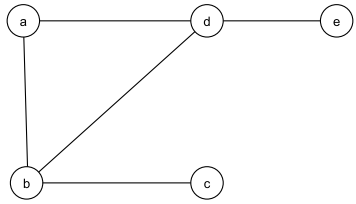

Figure 2.2 – Undirected graph $G = (V, E)$ with $V = \{a, b, c, d, e\}$ and $E = \{\{a, b\}, \{a, d\}, \{b, d\}, \{b, c\}, \{d, e\}\}$

Definition 2.1.3. The **order** of a graph is its number $|V|$ of vertices.
The **size** of a graph is its number $|E|$ of edges.

In directed graphs, edges are ordered pairs rather than pairs.

Definition 2.1.4. A directed graph is a pair $G = (V, E)$ with V is a non empty finite set of elements called vertices and E is a set of elements of V, called edges.
Let $V = \{x_1, x_2, ..., x_n\}$ and $E = \{(x, y), (x, y) \in E^2 \text{ and } (x, y) \text{ is an edge}\}$
$(x, y) \in E$ if and only if there is an edge from x to y.
x and y are neighbors or adjacent if (x, y) is a edge of G.

The edges of oriented graphs are also called arcs or directed links.

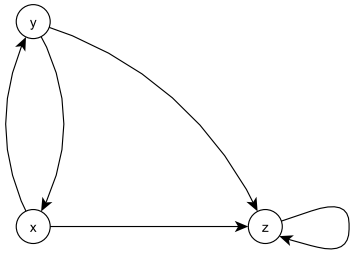

Figure 2.3 – Directed graph $G = (V, E)$
with $V = \{x, y, z\}$ and $E = \{(x, y), (x, z), (z, x), (z, y), (y, y)\}$

In this thesis, we will generally consider undirected graphs, except in section 3.3.2 where we study a 1D voter model. Whenever we talk about a graph $G = (V, E)$, by default this means, it is an undirected graph.

Definition 2.1.5. Let be $G = (V, E)$ a graph.
The degree of a vertex x, noted $d(x)$ or $d_G(x)$ is the number $|E_x|$ of edges at x.

We can represent a graph G by the lists of the neighbors of each vertex or by a matrix called the adjacency matrix of G.

Definition 2.1.6. The adjacency matrix of the graph $G = (V, E)$ with $V = \{x_1, x_2, ..., x_n\}$ is a Boolean matrix M defined by :

$$m_{i,j} = \begin{cases} 1 & \text{if } \{x_i, x_j\} \in E \\ 0 & \text{otherwise} \end{cases}$$

In the case of a complex system whose topology is represented by a graph $G = (V, E)$, we study the behavior of the states of each vertex (or agent) over time. Let $s_i(t)$ be the state of agent i at time t. A frequent case, this state depends both on the graph (topology) and on the states of all the vertices at previous time. In the discrete-time case, the system dynamics may be characterized by a transition function f_G (stationary state-space model) such that

$$\begin{cases} X(t+1) = f_G(X(t)) \\ X(0) = X_0 \end{cases} \tag{2.1.1}$$

where $X(t) = \begin{pmatrix} s_1(t) \\ \vdots \\ s_n(t) \end{pmatrix}$

The equation 2.1.1 is called the global state equation of the system. It enables us to study the behavior of the system. It defines the system dynamics, so that we can study the system behavior, run simulations, observe the system, find a way to control it, i.e. bring the system to a desired state, and determine its transient regime.
In some cases, the state equations are only defined "locally" by giving an agent state evolution as a function of the states of its neighbors. In this case, the dynamics of the system is be characterized for each agent i by a function f_i such that

$$s_i(t+1) = f_i\Big((s_j(t))_{j \in N_i}\Big) \tag{2.1.2}$$

where N_i is the set of neighbors of the agent i .
For instance, a node (or agent) state can be the measurement of electrical activity in an

electroencephalogram in the brain areas. For the epidemic spread model, the possible state value for each agent can be susceptible, infected or recovered. In economic models, states can be parameter values, such as a country's GDP, the value of a stock, etc. The states in the world 3 model are the values of all the data taken into account in the model: population, agricultural production, industrial output, persistent pollution index, Industrial Capital Output Ratio, etc.
The topology of the graph itself may evolve with time: edges and vertices may appear or disappear.

In this case, the system's dynamics is described with the dynamical graph: $G(t) = (V, E(t))$ and the global state equation:

$$X(t+1) = f(X(t), G(t)) \tag{2.1.3}$$

In section 4.3, we will present a way of detecting changes in system topology (appearance and disappearance of edges), by observing the system behavior.

In the case of a large graph, it may be necessary or desirable to determine a simplification of the system dynamics (complexity issues, lack of data). This problem is related to graph agregation and system reduction. It will be seen in section 5.2.
Graph theory offers also helps to solve some control and observability. In the following chapter, we present some results related to control theory and graphs.

2.1.3 Concepts about graphs

Generalities about graphs

As we'll see later, the behavior of the dynamics of a complex system represented by a graph is highly dependent on the structure of that graph. Thus, it is important to know the graph topology. We will investigate two particular topologies: the complete graph and the circular graph (see figure 2.4), for the sake of illustration.

Definition 2.1.7. Let (V, E) a graph with $V = \{x_1, x_2, ..., x_n\}$.

1. (V, E) is complete if $\forall (x, y) \in V^2$, such as $x \neq y$ $\{x, y\} \in E$ i.e. all possible edges (but self-edges) are in E. We note K_n a complete graph of order n.

2. (V, E) is circular if $\forall i \in [\![1; n-1]\!]$, $\{x_i, x_{i+1}\} \in E$ and $\{x_1, x_n\} \in E$.
   We note C_n a circular graph of order n.

In figure 2.4, we can see a complete graph and a circular graph with $n = 5$.

Figure 2.4 – K_5 and C_5

It's easy to imagine that, for similar local dynamics, the global dynamical behavior will be very different for these two graphs. For example, with the dynamic defined by:

$$s_i(t+1) \equiv \sum_{j \in N_i} s_j(t) \mod 2 \tag{2.1.4}$$

if the initial state of the system is $(s_1(0), s_2(0), ..., s_5(0)) = (1, 1, 0, 0, 1)$, we have:

- for K_5, $\forall t \geqslant 1,\ (s_1(t), s_2(t)..., s_5(t)) = (0, 0, 1, 1, 0)$
- for C_5, $\forall t \in \mathbb{N}$, $\begin{cases} (s_1(3t+1), s_2(3t+1), ..., s_5(3t+1)) = (0, 1, 1, 1, 1) \\ (s_1(3t+2), s_2(3t+2)..., s_5(3t+2)) = (0, 1, 0, 0, 1) \\ (s_1(3t+3), s_2(3t+3)..., s_5(3t+3)) = (0, 0, 1, 1, 0) \end{cases}$

We can see that in these two cases, with the same local dynamics (which implies only the sum of neighbors state) the behavior is different.

2.1.4 Connectivity, paths, distance and components

For observability and controllability, the notions of paths and connectedness are important.

Definition 2.1.8. A **path** of a graph $G = (V, E)$ is a sequence $(\alpha_0, e_1, \alpha_1, e_2..., e_k, \alpha_k)$ where $k \in \mathbb{N}^*$, $(\alpha_0, \alpha_1, ..., \alpha_k)$ is a sequence of elements of V, $(e_1, e_2, ..., e_k)$ is a sequence of elements of E, and $\forall i \in [\![1; k]\!], e_i = \{\alpha_{i-1}, \alpha_i\}$
The length of this path is the integer k. We can note $\ell(\alpha_0, e_1, \alpha_1, e_2..., e_k, \alpha_k) = k$
A path is a walk if the vertices $\alpha_0, \alpha_1, ..., \alpha_k$ are two by two distinct.
The **distance** between two vertices x and y is the minimum length of the paths connecting these two vertices. We note this integer $d(x, y)$.

Now, we can define a connected graph:

Definition 2.1.9. G is a **connected graph** if between any pairs of vertices, there exists a path.

This concept of connected graph allows us to define the notion of connected component in a graph.

Definition 2.1.10. Let $G = (V, E)$ be a graph. A component G_1 of G is a maximal connected subgraph in G which is maximal with respect to the inclusion order (there is no connected subgraph strictly containing G_1).

Remark 2.1.11. In a connected graph G, the only connected component is G.

For any graph, there is a connected subgraph partition. If a complex system is represented by a non-connected graph, we will not be able to control the system from a single vertex. In this case, it is necessary to consider a local control for each subgraph of the partition independently. For this reason, in this thesis, we will only consider connected graphs.

Definition 2.1.12. Let be $G = (V, E)$ a connected graph, the **diameter** of G is a positive integer, defined by $\delta(G) = \max_{(x,y)\in V^2} d(x, y)$

The diameter plays an important role in control systems, because the influence of one node on the others depends on the distance separating it from them. The diameter is therefore an important parameter in control cost.
On the other hand, the topological study of a graph makes it possible to identify those agents who have a strategic location and should have the greatest influence.

Graph metrics

We are now going to define some measure of vertices relative importance (see e.g. Newman's book "Networks" [76] for details).

Definition 2.1.13. The Clustering coefficient for node i is defined as:

$$C_i = \frac{\text{(number of pairs neighbors of i that are connected)}}{\text{(number of pairs of neighbors of i)}} \tag{2.1.5}$$

We can write:

$$C_i = \frac{2|T_i|}{k_i(k_i - 1)} \tag{2.1.6}$$

with $k_i = d(i)$ is the degree of i, $T_i = \{\{i, j, k\}, \big(\{i, j\}, \{j, k\}, \{k, i\}\big) \in E^3\}$

The clustering coefficient of a node, is a measure of the interconnections of its neighbors.

Definition 2.1.14. In a connected graph with n nodes, closeness centrality is a measure of the inverse of the average shortest distance from each vertex to each other vertex.
Let $G = (V, E)$ a connected graph, the closeness centrality of vertex i is the real value $c_c(i)$ defined by :

$$c_c(i) = \frac{n}{\sum_{j=1}^{n} d(i,j)} \tag{2.1.7}$$

This quantity enables us to identify the vertices that have a central position in the graph. This position has a role to play in the influence that an agent can have on the system. Identifying central vertices is therefore an important element in controlling a system, as the influence of each vertex on the others depends on the distance between them. But this is not the only element to be studied: the number of neighbors is also important, (i.e. the degree), and is a criterion for choosing which vertices to force to achieve a desired system state. The number of routes through a given vertex also plays an important role in this choice. It is quantified by betweenesss centrality.

Definition 2.1.15. The Betweenness centrality of a node k is a real value $\Delta(k)$ defined by

$$\Delta(k) = \sum_{i \neq j, i \neq k, j \neq k} \frac{\sigma_{ij}(k)}{\sigma_{ij}} \tag{2.1.8}$$

where σ_{ij} is the number of shortest paths between i and j, and $\sigma_{ij}(k)$ is the number of these paths that pass through k.

2.1.5 Community, modularity

The notion of community is an important concept when studying a complex system modelled by a graph. Communities are of interest to simplify a large system (reduction or coarse graining), and to suggest a strategy when trying to control a system. For dynamical systems, the knowledge of the community structure is very useful: it makes it easier to study the behavior of the network (for instance to control or to observe it) because we can decompose these problems into smaller problems defined on the communities subgraphs. Despite its importance, there is non unique and formal definition of community. In some cases, the communities are not disjoint two by two. For example, in social networks, a person can belong to different communities. In other cases, the set of communities is a partition of the set of vertices in the graph. This is the case we will consider in this thesis. In the literature (for example [117], [10], [14]), a community is often (informally) defined as a dense sub-graph weakly linked to the rest of the graph. This idea is illustrated in figure 2.5 for the partition of a graph into 7 disjoint communities.

Figure 2.5 – Example of a graph partitioned into 7 communities. Nodes of the same color belong to the same community

For a given graph, there are several possible community partitions. To evaluate these different partitions, we need a function that assesses their quality. The most widely used quality function is the modularity.

Definition 2.1.16 (Newman and Girvan [74])**.** Let $G = (V, E)$ be a graph and M its adjacency matrix, the modularity of a partition $\mathcal{C} = \{c_1, c_2, ..., c_p\}$ is the real value Q defined by:

$$Q = \frac{1}{2m} \sum_{i,j} \Big(m_{i,j} - \frac{k_i k_j}{2m} \Big) \delta(c_i, c_j) \tag{2.1.9}$$

where m is the total number of edges of the graph, $m_{i,j}$ is the coefficient (i, j) of the adjacency matrix, k_i is the degree of agent i, $c_i = \{1, 2, ...p\}$ is a label specifying the community of node i, p is the number of communities and δ is the Kronecker symbol, we have

$$\delta(c_i, c_j) = \begin{cases} 1 & \text{if } c_i = c_j \text{ i.e. } i \text{ and } j \text{ are in the same community} \\ 0 & \text{otherwise} \end{cases}$$

This notion of modularity is a mean of giving a formal definition of communities in the case where the set of communities is a partition of vertices, and when the number of communities is fixed. In this case, we can say that the p communities are the p

subsets that form a partition of the set and maximize the modularity. But there are other possible definitions of communities. Fortunato [34] identifies three variants of definitions: the locally oriented definition, the globally oriented definition and the one based on node similarity. In the first variant, the intrinsic quality of each community taken separately and independently of the whole graph is evaluated. The second variant assigns a quality criterion to the entire graph to check that the set of communities is coherent as a whole. The third variant is based on the notion of node similarity. To determine communities, we will use the second approach because it's well suited to the scale-free graphs we will be considering (these graphs have hubs).

The determination of communities has taken on a very important role in research with the development of social networks. But, as there is no formal universal definition of communities, there are many different ways of determining them.

Girvan and Newman [39] and Raddicchi [88] proposed an algorithm that uses edge betweenness as a metric to identify the boundaries of communities. Boccaletti et al. [11] show how to combine topological and dynamic information in order to identify the community structure of a graph. Karrer et al. [51] propose a method to measure the robustness of a community structure with small perturbations.
In section 4.4, we present an original method for determining communities by studying the influence of the nodes on the system, without knowing the interaction topology in the system. In our case, we want to define a network partition to facilitate system control. Consequently, the communities we build depend on the dynamics of the system. This means that the notion of community we're considering is somewhat different from conventional definitions. So we will use a different name: community of influence. We will see that these notions of communities can coincide for certain dynamics.
Most methods for constructing a community partition are static and use only the topology of the system, such as the Louvain method [10] which maximizes modularity. We propose a different approach, which in some respects resembles the random walk method based on Markov stability described in the papers [83], [27], [17]. In both cases, communities are determined using the dynamics defined on the network.

One aim of the partitioning of a system into communities is to obtain simplified system dynamics. The concept of a graph community is expected to enable us to reduce the system by agregation and consider each community, which are generally of different sizes, as a supernode. These supernodes are obtained by merging the nodes of each community. We define a new graph where the new nodes are these supernodes, and we define the states of these supernodes as functions of the states of the nodes in these communities. This involves determining the dynamics of the simplified system for the states of these supernodes. The reduction may greatly reduce the computational cost, but introduce an error between the real system and the simplified system, which we

need to be able to evaluate. In the section 5.2, we look at this problem.

2.1.6 Graph representation for complex systems

There are several approaches to generate graphs that can model complex systems. In this section we will present three specific networks: the Erdös-Rényi random networks, small world networks and Barábasi scale-free networks. We are only going to present the intuitive ideas behind models which generate these three kind of graphs. First, we will present the model established in 1959 by Paul Erdös and Alfréd Rényi. We will introduce this model because it was one of the first random graphs to appear in the 50s and has been the most widely studied. This network model is very simple. In many branches of the social sciences it is necessary to have more elaborate models. We will introduce therefore the Watts and Strogatz's small-world graphs. These small-world networks have their origins in an experiment by Milgram, whose aim was to demonstrate that everyone can be linked to any other individual by a short chain of social relations. Finally, we'll look at another type of model: preferential attachment graphs. The principle behind these randomized graphs is that as the graph grows, each node added is preferentially linked to nodes of greater degree.

Erdős-Rényi model

This first theoretical model was defined by Paul Erdős and Alfréd Rényi [31] in 1960. This model is characterized by a graph $G_{n,p}$ with n is the number of vertices, and p is the probability that each pair of vertices $\{x_i, x_j\}$ is an edge. With this construction, we obtain a uniform distribution of edges between pairs of graph vertices. The degree of each vertex thus follows a binomial distribution of parameter (n, p).

Small world graph of Watts and Strogatz

A small-world network is a type of graph in which most nodes are not direct neighbors of one another, but most nodes can be reached from every other by a small number of hops [110]. The concept of small world comes from Milgram's work [68] in the 60's. Milgram was a social psychologist who took up the concept of the 6 degrees of separation invented by a Hungarian writer. This theory states that each of us on earth can be connected to another person by following a chain of acquaintances containing no more than 6 people. Milgram conducted several social experiments to measure the diameter of social networks. His work led to the conclusion that the world was made up of several small worlds.
Watts and Strogatz published a paper [110] in 1998 that proposes a model for small world graphs. This model represents social networks where close relationships are

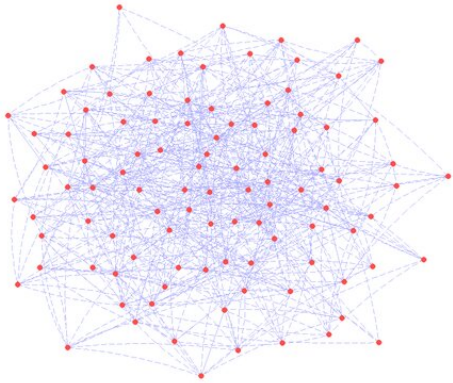

Figure 2.6 – Example of an Erdős-Rényi graph with $n = 100$ and $p = 0.1$

privileged, i.e. each person has mainly social links with geographical neighbors, and a few more distant social links. A graph is said to be a small world if the shortest path between two nodes has an average logarithmic length equal to the number of vertices.

Watts and Strogatz small world graphs are defined from 3 parameters (n, m, p). The natural numbers n and m are respectively the order of the graph and the half order of each node's small world, and p is the probability of redeployment of each initial edge ($p \in]0; 1[$).

The original Watts and Strogatz model starts with a ring graph of n vertices in which each vertex is symmetrically connected to its $2m$ nearest neighbors. Then, the connections previously created between neighbors are rewired with probability p, to connects to a randomly chosen vertex, avoiding self-connections, and typically creates shortcuts between distant parts of the ring. The parameter p therefore tunes the level of randomness present in the graph, keeping the number of edges constant. With this construction, the rewiring process does not change the average degree which is $k = 2m$. According to Watts and Strogatz, only a very small percentage of random "shortcuts" is needed for this model to be a small world. So we need to choose a value of p small enough to have few shortcuts. Figure 2.7 shows the different steps of construction of the small world model of Watts and Strogatz. There are many other models that give small world graphs, for example Barábasi and Albert's model, which we will now describe.

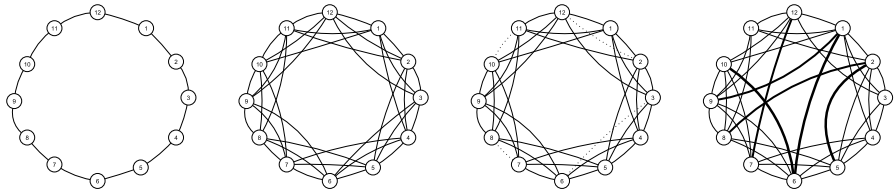

Figure 2.7 – Generation of a small world graph with a Watts-Strogatz model. **Step 1 :** generate a ring graph. **Step 2:** add the vertices that connect the nodes with the $2m$ nearest neighbours. in this case, $m = 3$, for exemple for the node 5, it is already connected to nodes 4 and 6, we add the links with $2, 3, 7$ and 8. **Step 3:** for each existing edge (i, j), it is removed with a probability p. Rewiring is done by replacing (i, j) with (i, k) where k is chosen uniformly at random from all possible nodes while avoiding self-loops.

Scale free graph

In the 1970s, Price became interested in the citation network of scientific papers. He proposed a model in 1976 [87]. Price's idea was that the number of new citations a paper receives was proportional to the number of citations it had previously received. The same behavior can be observed for other growing networks, such as the World Wide Web. This model was a growing network of papers represented by a directed graph with a preferential attachment. The nodes of the graph are the published papers, and (a, b) is an arc when a has cited b. The result is a graph with preferential attachment, if a new paper will tend to preferentially quote papers with the highest incoming degree. The model obtained from this study was a network represented by a directed graph with a preferential attachment.

Price's model generates a power law degree distribution. There is a strictly positive real number λ such that $P(k) \sim k^{-\lambda}$, where $P(k)$ is the distribution of degree. It is obtained by first counting the number of edges adjacent to each node, and then dividing this number by the total number of nodes (frequencies of nodes of degree k). When the degree distribution of a graph follows a power law, it is said to be a scale-free graph. The description of this model can be found in Mark Newman's book Networks [72].

In 1990s Barábasi and Albert [5] have proposed a model based on the same idea of preferential attachment as Price's, except that the edges are non-oriented. In their model, nodes are added one by one, and each node preferably connects to a set of existing nodes. This model is represented by an undirected graph and the degree distribution is a power of degrees with $\lambda = 3$ (the proof of this result is in the Newman's

book [72]). The Barábasi-Albert model are reference network with hubs and quite often a good representation of social networks.
These scale free graphs are defined from 3 integer values (n, k, m). The number n is the order of the graph, k is the order of the initial graph and m is the number of links created for each node added when we generate the graph. To obtain a scale free graph of Barábasi and Albert, in the beginning, we start with complete graph of k nodes.
Then we add the vertices one by one. Each new vertex is connected to m existing vertices with a probability that is proportional to the degrees of the existing vertices. The probability p_i that the new node is connected to node i is therefore:

$$p_i = \frac{k_i}{\sum_j k_j} \tag{2.1.10}$$

where k_i is the current degree of the node i. In figure 2.8, we can see a description of the different steps of the construction of this scale free graph, and in figure 2.9, we have an example of a Barábasi and Albert graph with $n = 100$. In the section 3.2, , we use this model to represent the social network of a voter model (see Fig. 3.2).

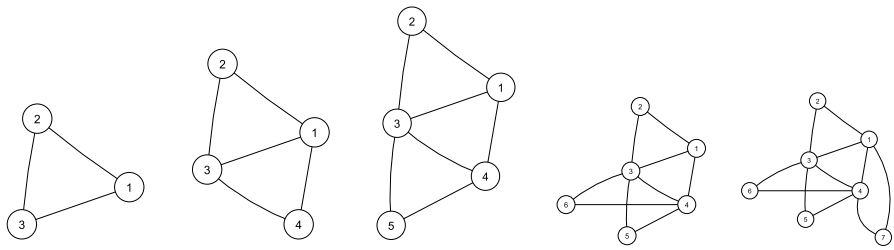

Figure 2.8 – Generation of a scale free graph. **Step 1:** we build a complete graph of order k, here $k = 3$. **Step 2:** we add a vertex, we connect it to m existing nodes with probability $1/k$. Here $m = 2$ **Step** 3**,** 4**, ...** n**:** we add a vertex, we connect it to m existing nodes with probabilities $p_i = k_i/(\sum_j k_j)$ where k_i is the current degree of node i. In this case, when we add the edge 5, we have $p_1 = 3/10, p_2 = 2/10, p_3 = 3/10,$ and $p_4 = 2/10$.

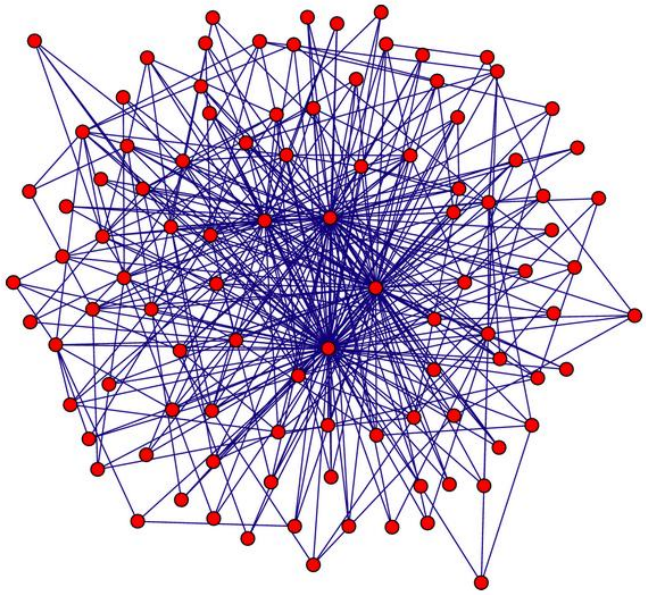

Figure 2.9 – Example of scale free graph with $N = 100$, $k = 4$ and $m = 3$

2.2 Reachability and observability of control complex systems

2.2.1 Introduction

Control theory is a branch of applied mathematics concerned with the control of dynamical systems. It studies the possibility of acting on a time-dependent dynamic system, in order to duve it at a desired configuration. Controllability and observability were introduced by Kalman [48] , [47] in the case of a finite-dimensional linear systems, and his results are now the basis of control theory. There are many fields of application for control theory, including physics, chemistry, thermodynamics, aeronautics, computer science, biology, medicine, economics, etc. The aim of this chapter is to present some general principles of control theory. We will start first with a general study of linear systems, then we will consider the special case of large scale systems (linear or non-linear). We will introduce the subject with continuous-time systems, which are the most usual use in control, and then concentrate on discrete-time systems, having exclusively used this type of model in this thesis. We will start by defining the notions of controllability, reachability and observability. Then we will

present some general results for linear discrete-time systems. Finally, we will look to the problem of large-scale systems, for which it is difficult to apply the general principles of control theory, the complexity of the calculations and the size of the systems being limiting factors. Antoulas [4] became interested in this problem: he introduced an approximation method for model reduction of large-scale dynamical systems. This is a projection method which combines aspects of the Singular Value Decomposition (SVD) and Krylov based reduction methods. SVD is a three-matrix factorization based on the algebraic and geometric properties of linear transformations. Kryvlov's method allows us to find solutions to the controllability and observability problems presented in this chapter using properties of linear algebra. To control a large system, Siljak [100] has a different approach based on linear algebra. Using structural results from networks: he proposed to decompose it into sub-systems, and from these determines a decentralized control method.

2.2.2 General principles of control theory

Mathematical system modeling

Many dynamical systems are described by a differential equation of the form:

$$\begin{cases} \dot{x}(t) = F(t, (x(t)) \\ x(t_0) = x_0 \end{cases} \tag{2.2.1}$$

with
$x(t)$, the state vector of the system.
$\dot{x}$, the time derivative the time of the state vector $x(t)$
x_0, the initial state vector.
F, the model, a vector field, which explicitely depends on t when the dynamics can change over time. Let $\mathbb{X}$ be the state vector space, $\forall t, x(t) \in \mathbb{X}$.
System 2.2.1 is an uncontrolled system. In control theory, we wish to act on the system. The equation representing the system dynamics becomes:

$$\begin{cases} \dot{x}(t) = F(t, (x(t), u(t)) \\ x(t_0) = x_0 \end{cases} \tag{2.2.2}$$

where, the vector $u(t)$ denotes the control at time t. We select $u(t)$ to modify the state vector $x(t)$ and orient it towards a desired state.
In the discrete-time case, the system can be represented by the following equations:

$$\begin{cases} x(k+1) = f(x(k), u(k)) \\ x(0) = x_0 \end{cases} \tag{2.2.3}$$

with $x(k) \in \mathbb{R}^n$ is the state vector and $u(k) \in \mathbb{R}^m$ is the control input.

Reachability, Controllability, and Observability

In this section, we present the concepts of controllability, reachability and observability. We will start by defining the notion of energy, which is an important concept in control theory: it allows us to quantify the cost of controlling a system, to quantify the information we can obtain about system states from observations.

Definition 2.2.1. In the continous-time system case, let f a function from $I \subset \mathbb{R}$ to $\mathbb{K}$ with $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$.
We say that f has finite energy if $f \in L^2(I, \mathbb{K})$ and in this case we can define the energy of f as the real number $||f||^2$ defined by

$$||f||^2 = \int_I |f(t)|^2 dt \tag{2.2.4}$$

For a discrete-time system, let $(x_k)_{k\in J}$ with $J \subset \mathbb{N}$ a sequence, we say that (x_n) has finite energy if $(x_k)_{k\in J} \in \ell^2(J, \mathbb{K})$ i.e. $\sum |x_k|^2$ is a convergent series and the energy is the real number denoted $||x||^2$ defined by:

$$||x||^2 = \sum_{k\in J} |x_k|^2 \tag{2.2.5}$$

For the definition of reachability, controllability, and observability we follow Antoulas [4].

Reachability Let Σ be a dynamical system whose dynamics is defined by 2.2.2.
Let $\phi(u, x_0, t)$ denote the state of the system reached at time t from the initial state x_0, under the influence of the input $u(\tau)$ for $\tau \in [0, t]$.
Let $\mathbb{X}$ be the state place.

The problem of **reachability** consists in finding the set of all final states x_1 reachable from a given initial state x_0 at time t_0 by with the appropriate input u.

Definition 2.2.2. A state x_1 is reachable from the state x_0 if there exists an input function $\overline{u}$ of finite energy, and a time t_1 finite such that:

$$x_1 = \phi(\overline{u}, x_0, t_1)$$

The reachable subspace $\mathbb{X}^r \subset \mathbb{X}$ is the set containing all reachable states of Σ.
The system Σ is completely reachable from x_0 if $\mathbb{X}^r = \mathbb{X}$.

According to the definition 2.2.1, for $\overline{u}(t)$ to have finite energy, we must have $u \in L^2([t_0; t_1]$.

For discrete-time systems, we have the following similar definition

Definition 2.2.3. For the system Σ defined by the equation 2.2.3, a state x_1 is reachable from the state x_0, if there exist a time n and input control vectors

$$u = (u(0), u(1), ..., u(n-1))$$

such that

$$x_1 = \phi(u, x_0, n)$$

where $\phi(u, x_0, n)$ denote the state of the system at time t reache from the initial state x_0, with the input $(u(0), u(1), u(2), ..., u(n-1)$.

System Σ defined by the equation 2.2.3 is said to be reachable from x_0 if any state is reachable from x_0.

Controllability The problem of **controllability** consists in finding the set of all initial states x_0 that are controllable to a final state x_1.

Definition 2.2.4. A state $x_0 = x(t_0)$ ($x_0 \in \mathbb{X}$) is controllable to the state x_1 if there exist an input function $\overline{u}(t)$ of finite energy, and a time t_1 finite such as :

$$\phi(\overline{u}, x_0, t_1) = x_1$$

The controllable subspace $\mathbb{X}^c \subset \Sigma$ is the set of all controllable state.
The system Σ is completely controllable

$$x_0 \text{ and } x_1$$

For discrete-time systems, we have:

Definition 2.2.5. For a system Σ defined by the equation 2.2.3, a state $x_0 = x(0)$ is controllable to the state x_1, if there exist a time n and input sequence

$$u = (u(0), u(1), ..., u(n-1))$$

such that

$$\phi(u, 0, n) = x_1$$

Definition 2.2.6. System Σ defined by the equation 2.2.3 is said to be controllable to x_1 if any state is controllable to x_1.

Remark 2.2.7. The notion of reachability is stronger than that of controllability. In the continuous case, these 2 notions are equivalent (see [4]).

Observability Since the state of the system is often not directly measurable or observable, we introduce the notion of an output equation.

$$y(t) = C(x(t)) \tag{2.2.6}$$

where $y(t)$ denotes the vector of "measured" values with usually depend on the state vector $x(t)$.
Typically, we put some sensors that allow us to study the system through measurement.

Definition 2.2.8. The dynamical system Σ defined by the equation 2.2.2 is said to be observable if, for any state $x_0 \in \mathbb{R}^n$, there exists a finite time t_1 and a control $u \in L^2([t_0; t_1])$ such that knowledge of $y(t)$ for $t \in [0; t_1]$ enables the initial state x_0 to be determined.

We have $x_1 = \phi(u, x_0, t_1)$. The observation problem reduces to given $C(x_1)$, we can find x_0 For a discrete-time linear system, we have:

Definition 2.2.9. The dynamical system Σ defined by the equation 2.2.3 is said to be observable if, for any state $x_0 \in \mathbb{R}^n$, there exists a finite time n such that knowledge of $y(n)$ enables the initial state x_0 to be determined.

2.2.3 Linear dynamical system

In the linear case, the system dynamics are described by the following differential system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \\ x(0) = x_0 \end{cases} \tag{2.2.7}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, and $C \in \mathbb{R}^{p \times n}$ with $(n, m, p) \in \mathbb{N}^3$ such as $m \leqslant n$ and $u \in L^2([0; T], \mathbb{R}^m)$.

In this thesis, we will focus on discrete-time systems[1]. The following equations describe the state-space model of a discrete-time linear control system Σ:

$$\begin{cases} x_{k+1} = Ax_k + Bu_k \\ y_k = Cx_k \\ x_0 \in \mathbb{R}^n \end{cases} \tag{2.2.8}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$ $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $u_k \in \mathbb{R}^m$ et $C \in \mathbb{R}^{p\times n}$
Thus the linear dynamical system Σ is characterized by the matrices A, B and C. We note $\Sigma = (A, B, C)$. Σ is represented in figure 2.10 together with an output feedback linear controller K.

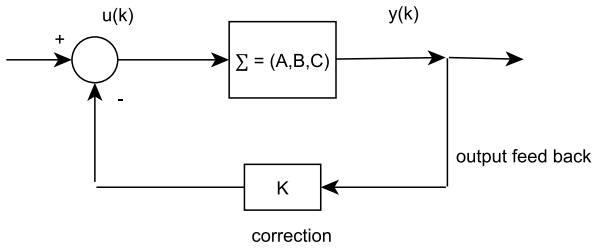

Figure 2.10 – Schema of a linear system $\Sigma = (A, B, C)$. K is used to modify the control according to observed values to achieve a desired state: $u(k + 1) = Ky(k)$

The state of the system x_k corresponding to the control $(u_i)_{0\leqslant i\leqslant k-1}$ is:

$$x_k = A^k x_0 + \sum_{i=0}^{k-1} A^{k-1-i} B u_i \tag{2.2.9}$$

From the equation 2.2.9, we can define the controllability matrix:

Definition 2.2.10. The controllability matrix of the system 2.2.8 is the matrix $\mathcal{C}_n \in \mathbb{R}^{n\times n^2}$ defined by:

$$\mathcal{C}_n = [A; AB; A^2B; \ldots; A^{n-1}B]$$

Theorem 2.2.11 (Kalman's criterion of controllability)**.** The system is controllable if and only if the rank of the controllability matrix is maximal, i.e. $rank(\mathcal{C}_n) = n$

[1] In the book [26] the link from the continuous-time equation system to the discrete-time case is developed.

When the system 2.2.8 is controllable, we also say equivalently that the pair (A, B) is controllable.
Controllability means that we have sufficient freedom to influence the system's behavior and achieve any desired state by choosing input values appropriately.

Observabilility We consider the discrete-time linear control system represented by the system 2.2.8. According the equations 2.2.8 and 2.2.9, we get:

$$y_k = CA^k x_0 + \sum_{i=0}^{k-1} CA^{k-1-i} B u_i \tag{2.2.10}$$

Definition 2.2.12. The observability matrix of the system 2.2.8 is the matrix $\mathcal{O}_n \in \mathbb{R}^{nm \times n}$ defined by:

$$\mathcal{O}_n = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Theorem 2.2.13 (Kalman's criterion of observability)**.** The system is observable if and only if the rank of the observability matrix is maximal, i.e. $rank(\mathcal{O}_n) = n$

When the system 2.2.8 is observable, we also say that the pair (A, C) is observable.

The Kalman's criterion allow to check wether the systems is controllable (resp. observable) or not. However, it does not account for how much effort is needed to do so. For example, Razakanirina and Chopard [89] modelled a credit market and showed that during a period of market instability, it is possible to return to a certain stability, but this requires:

- either to wait a long time.
- or exponential growing amounts of money.

To quantify this control cost, we may use the Grammian that we are going to define now.

Grammians

To define the Grammians, we need define the notion of a stability.

We are now going to define particular state vectors that are an important role in the study of control systems: steady states or equilibrim state. To determine these vectors, it is necessary not to excite the system, so they are defined with a constant control vector:

Definition 2.2.14. The steady state defined with a constant control u_c is the state vector X_{eq} that satisfies the fixed point equation:

$$X_{eq} = AX_{eq} + Bu_c$$

Definition 2.2.15. The system 2.2.3 is stable if exists a state $\overline{x}$ such as:

$$\forall \epsilon > 0,\ \exists \eta > 0,\ ||x(0) - \overline{x}|| < \eta \Rightarrow \forall k \in \mathbb{N},\ ||x(k) - \overline{x}|| < \epsilon$$

The system 2.2.3 is asymptotically stable if a state $\overline{x}$ exists such as $\lim\limits_{k \to +\infty} x(k) = \overline{x}$, i.e.:

$$\forall \epsilon > 0,\ \exists k_0,\ \forall k \geqslant k_0,\ ||x(k) - \overline{x}|| < \epsilon$$

The equilibrium state is globality asympotically stable if for all initial state x_0, $\lim\limits_{k \to +\infty} x(k) = \overline{x}$

The following theorem characterizes this notion of asymptotic stability for linear discrete-time system.

Theorem 2.2.16. The system 2.2.8 is asymptocally stable if all the eigenvalues of the matrix A are strictly less than 1 or less than 1 and any eigenvalues on the unit circle have multiplicity one.

This theorem is a consequence of the equation 2.2.9.

We define hereafter observability and controllability Grammians for asymptotically stable linear discrete-time system of the form 2.2.8.

Definition 2.2.17. The Controllability Grammian W^c and the Observability Grammian W^o, also called grammian matrices, are respectively defined by

$$W^c = \sum_{k=0}^{+\infty} \left(A^k B B^T (A^T)^k\right) \tag{2.2.11}$$

$$W^o = \sum_{k=0}^{+\infty} \left((A^T)^k C^T C A^k\right) \tag{2.2.12}$$

The Controllability Grammian W^c is related to the minimum energy of the control signal (u_k) required to stear the state from an initial state $x_0 = 0$ to a final state x_f on an infinite time horizon.
This minimum energy of the input sequence (u_k) may be computed as:

$$||u||^2 = \lim_{n\to\infty} \sum_{k=0}^{n} |u_k|^2 \tag{2.2.13}$$

$$= \lim_{n\to\infty} \sum_{k=0}^{n} u_k^T u_k \tag{2.2.14}$$

$$= x_f^T (W^c)^{-1} x_f \tag{2.2.15}$$

On the other hand, the energy for some output signal (y_k) is:

$$||y||^2 = \lim_{n\to\infty} \sum_{k=0}^{n} |y_k|^2 \tag{2.2.16}$$

$$= \lim_{n\to\infty} \sum_{k=0}^{n} y_k^T y_k \tag{2.2.17}$$

$$= \lim_{n\to\infty} \sum_{k=0}^{n} x_0^T (A^T)^k C^T C A^k x_0 \tag{2.2.18}$$

$$= x_0^T W^o x_0 \tag{2.2.19}$$

From this energitical interpretation, it may be seen that Grammians provide both necessary and sufficient conditions for observability or controllability, and, in the same time indicate how much controllable or observable are the nodes in the state space.

Theorem 2.2.18. Assume system 2.2.8 is asymptotically stable. Then

- The pair (A, B) is controllable if and only if the controllability Grammian W^c is positive definite.
- The pair (A, C) is observable if and only if the observability Grammian W^o is positive definite.

Expressions 2.2.11 and 2.2.12 are difficult to use for calculating Grammians. The following theorem provide Lyapunov equations which may be solved to obtain numerically the Grammians:

Theorem 2.2.19. The controllability grammian W^c and the observability Grammian W^o are the respective semi-definite positive solutions of the Lyapunov equations:

$$-W^c + AW^c A^T = -BB^T \tag{2.2.20}$$

$$-W^o + A^T W^o A = -C^T C \tag{2.2.21}$$

Example

We now present an example of a discrete linear system. A shepherd manages a number of sheep in his herd. Let $x_a(k)$ be the number of hundred of sheep with age $a \in \{1, 2, 3+\}$
Each year, wolves decimate 30 percent of the flock, and 10 percent of the sheep aged 3 (and over) die of old age.
Sheep aged 1 year do not reproduce, sheep aged two years have an average of 0.6 lambs and those aged 3 years and over have an average of 0.4 lambs.
The shepherd buys or sells sheep aged 3 and over.
The control u_k is the number of hundred of sheep bought, or sold $(u_k < 0)$.
The output y_k is the total number of hundred of sheep in the herd.

The state of the herd is represented by the following system of equations:

$$\begin{cases} x_1(k+1) = 0.6x_2(k) + 0.4x_3(k) \\ x_2(k+1) = 0.7x_1(k) \\ x_3(k+1) = 0.7x_2(k) + 0.6x_3(k) + u(k) \\ y(k) = x_1(k) + x_2(k) + x_3(k) \end{cases} \tag{2.2.22}$$

Let $X(k) = \begin{pmatrix} x_1(k) & x_2(k) & x_3(k) \end{pmatrix}^T$.
At the initial moment, the shepherd has 100 sheep of each age, $X(0) = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}^T$.
The system of equations 2.2.22 can be written as:

$$\begin{cases} X(k+1) = AX(k) + Bu_k \\ Y(k) = CX(k) \end{cases} \tag{2.2.23}$$

with $A = \begin{pmatrix} 0 & 3/5 & 2/5 \\ 7/10 & 0 & 0 \\ 0 & 7/10 & 3/5 \end{pmatrix}$, $B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and $C = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$.
The controllability matrix is:

$$\mathcal{C}_2 = [A, AB, A^2B] = \begin{pmatrix} 0 & 3/5 & 2/5 & 2/5 & 6/25 \\ 7/10 & 0 & 0 & 0 & 7/25 \\ 0 & 7/10 & 3/5 & 3/5 & 9/25 \end{pmatrix}$$

The rank of this matrix is 3, so the system is reachable.
And the observability matrix is:

$$\mathcal{O}_2 = \begin{pmatrix} A \\ CA \\ CA^2 \end{pmatrix} = \begin{pmatrix} 0 & 4/5 & 2/5 \\ 9/10 & 0 & 0 \\ 0 & 9/10 & 3/5 \\ 9/10 & 17/10 & 1 \\ 153/100 & 81/50 & 24/50 \end{pmatrix}$$

Its rank is also 3, so the system is observable.
The eigenvalues of A are $\lambda_1 \simeq 0.973, \quad \lambda_2 \simeq -0.490$ and $\lambda_3 \simeq 0.117$.
as the eigenvalues are strictly less than 1, according the theorem 2.2.16 , the system is asymptotically stable.
By solving the Lyapunov equations 2.2.20 and 2.2.21, we obtain the controllability Grammian W^c and the observability Grammian W^o:

$$W^c = \begin{pmatrix} 1.745 & 1.154 & 2.390 \\ 1.154 & 0.855 & 1.569 \\ 3.390 & 1.569 & 4.276 \end{pmatrix} \tag{2.2.24}$$

$$W^o = \begin{pmatrix} 14.699 & 20.169 & 15.817 \\ 20.169 & 27.957 & 21.832 \\ 15.817 & 21.832 & 17.100 \end{pmatrix} \tag{2.2.25}$$

Using the equation 2.2.19, the energy by observing is:

$$E = ||y||^2 = X(0)^T W^0 X(0) = 175.392$$

Let's now consider that we can only observe the number of sheep in one age class of the flock. If it's the number of sheep aged 1 year, we have $C_1 = (1\ 0\ 0)$. Similarly we have $C_2 = (0\ 1\ 0)$ and $C_3 = (0\ 0\ 1)$ respectively for sheep aged 2 years, and those aged 3 years and over. For these 3 cases, we obtain the following Gramians and observability energy

$$W_1^o = \begin{pmatrix} 2.474 & 1.933 & 1.502 \\ 1.933 & 3.007 & 2.288 \\ 1.502 & 2.288 & 1.745 \end{pmatrix} \text{ and } E_1 = 14.699 \tag{2.2.26}$$

$$W_2^o = \begin{pmatrix} 1.212 & 0.947 & 0.736 \\ 0.947 & 2.474 & 1.121 \\ 0.736 & 1.121 & 0.855 \end{pmatrix} \text{ and } E_2 = 27.957 \tag{2.2.27}$$

$$W_3^o = \begin{pmatrix} 2.619 & 3.533 & 2.745 \\ 3.533 & 5.344 & 4.182 \\ 2.745 & 4.182 & 4.276 \end{pmatrix} \text{ and } E_3 = 17.010 \tag{2.2.28}$$

The results obtained suggest that it is preferable to know the total number of animals in the herd: in fact, this observation has the highest energy of the 4 proposed, which means that less effort will be required to find the number of animals in each age class. If only one of the 3 age classes is known, for the same reason, we'd rather observe the number of 2-year-old sheep.

2.2.4 Structural approach for control systems

The structural study of the system allows us to orientate the strategy to be followed to control a system, for instance in choosing actuators and sensors locations. The topological structure of the graph representing the links between its state variables input and output has an important role in this strategy. This approach to structural controllability was introduced by Lin in 1974 [56]. In this paper, he defines the concept of structural controllability and proposes some results in the linear case. Commault and Dion have proposed numerous results about this subject, including the following articles [22], [23], [28]. Liu, Slotine and Barábasi are also interested in the theory of structural controllability, including the case of non-linear systems [58], [59], [61], [60]. In particular, they emphasise the importance of knowing the degree distribution. Knowledge of the structure is all the more important when the parameters of the system and the values of A are not known, or when the system is non-linear. Indeed, in these cases, it is not possible to obtain the states over time or the computations may be complicated to obtain them. Knowledge of the graph structure allows their to determine the nodes that should be influential, as well as those that have a minor role in the dynamics of the system due to their position in the graph (regardless of the numerical values for model parameters).

Gao et al. [37] study the impact of topology on the effectiveness of target control[2]. They present methods for determining pilot nodes for control when the network structure is perfectly known. They show that degree heterogeneous networks are target controllable with higher efficiency than homogeneous networks and that the structure of many real-world networks are suitable for efficient target control. For directed tree networks, they prove that one node can control a set of target nodes if the path length to each target node is unique. This example confirms that knowledge of the graph structure plays a very important role in controlling a system.

Large system

In the literature, we find several definitions of large systems. These definitions are based on the variety of these systems and the objectives for their study (control, intrinsic characteristics of the system, etc.). According to Siljak [100], a "large system" is a system which, for conceptual or numerical reasons, requires decomposition into a number of interconnected sub-systems. For Mansouri et al. [62] a large system is a system that we choose to see as a set of interacting subsystems, for reasons of reliability, flexibility, robustness, cost, etc. Claveau [21] considers that a large system is characterized by a large number of dynamics, but also a large number of input and

[2] by target control, they mean efficient control of a preselected subset of nodes

output variables.
These systems are often described using an extended spatial representation.
There is no accurate and universal definition of a large system, among others du to the variety of topologies of these systems. In most cases, the global mathematical state space model is not convenient for solving control problems and it is necessary to implement a method of decomposition and/or reduction. We will briefly review two different approaches. The first is based on a structural study of the system in order to break it down into sub-systems. Following Siljak [99], [100],[101], a "large system" is then broken down into a finite number of interconnected sub-systems. This decomposition into sub-systems leads to decentralized approaches for control problems. The second approach proposes a reduction of the system without using information on the network topology. This approach reduces the dynamics. Antoulas [4] presents different methods using results from linear algebra, following this second approach.

Decentralized control and decomposition into interconnected subsystems Centralization and decentralization are two types of structure that can be found in government and management. Centralization of authority means that power and decision-making is reserved exclusively for a central authority. In automation, most approaches are based on a centralized control model. Siljak [99],[100] [101] has proposed a decentralized approach in the case of a large system . This approach requires a structural study of the system in order to break it down into sub-systems. In this decomposition, we can simplify the system by neglecting certain interactions between sub-systems, and by truncating certain links.
In the case of linear systems, we can consider the following two types of decomposition: models relating to dynamically decoupled systems and models relating to output-decoupled dynamic system, i.e. a system whose dynamic equations are coupled, but whose observations are strictly local. In the first case, if we have N sub-systems, the system is represented by the following equation system

$$\begin{cases} \forall i \in [\![1; N]\!],\ E_i x_i(k+1) = A_{i,i} x_i(k) + B_i u(k) \\ Y(k) = CX(k) \end{cases} \tag{2.2.29}$$

In the second case, we have a system whose dynamical equations are coupled but whose observations are strictly local. It is represented by:

$$\begin{cases} \forall i \in [\![1; p]\!],\ E_i x_i(k+1) = A_{i,i} x_i(k) + B_i u(k) + \sum_{j=1, j\neq i}^{N} A_{i,j} x_j(t) \\ Y_i(k) = C_i X_i(k) \end{cases} \tag{2.2.30}$$

With
$X_i \in \mathbb{R}^{n_i}$, the state vector of the sub-system i
$A_{i,i} \in \mathbb{R}^{n_i \times n_i}$, matrix of the dynamic of the sub-system i
$A_{i,j} \in \mathbb{R}^{n_i \times n_j}$, matrix of the interconnection of sub-systems i and j
$B_i \in \mathbb{R}^{m \times n_i}$,
$E_i \in \mathbb{R}^{n_i \times n_i}$,
$Y_i \in \mathbb{R}^{n_i}$, the control vector of the sub-system i
n_i is the size of the sub system i.
Once we have the subsystems, we can move on to a simpler form of local control.
One of the most fundamental problems in control theory is stabilization. In the case of decentralized control of interconnected systems, it is generally necessary for the overall closed-loop system to be stable, but also for the closed-loop subsystems to be autonomous. Furthermore, stability is desirable even when disturbances occur at the interconnections between the subsystems. This type of robust stability is called connective stability. Connective stability is then a technique that guarantees the local stability of each sub-system and the global stability of the overall system, while taking into account the interactions of the sub-blocks. The study of the connective stability of interconnected systems consists in ensuring the stability of the global system, which consequently ensures the stability of all the sub-systems.

Model reduction We will now discuss the second approach, which is based on a simplification of the system dynamics. For large dynamical systems, it is very difficult to define a control law. Numerical determination of a solution (even an approximate one) to a complex system requires significant computing ressources. Large size implies problems due to the limits of memory space, computing capacity and precision. Model reduction is therefore necessary. Reducing a model aims at limiting the number of variables while remaining faithful to the initial model. By reducing the size of the problems, we achieve a smaller computational complexity, more stable numerical calculation algorithms and, a fortiori, shorter simulation calculation times. Antoulas [4] was interested in this problem of system reduction.
According to him, there are three main classes of efficient reduction methods for large scale linear systems:

- those based on Singular Value Decomposition (SVD).
- those based on Krylov's method.
- mixed iterative methods combining certain aspects of a of the two previous ones.

Whatever numerical linear algebra methods are used, the general purpose of the reduction methods is to project the dynamics onto a low dimensional subspace in the state

space, while retaining the more controllable and observable modes (see, for instance balanced realizations and truncations methods in [4]).

This reduction is also relevant for non-linear systems, where the complex behavior of the system requires calculation algorithms whose convergence is not guaranteed. However, in the non-linear case, there are mostly empirical methods such as PDD.

It is important to define the projection that will allow the least information to be lost. Antoulas suggests using the orthogonal eigenvalue decomposition method, known as principal component analysis and singular value decomposition (SVD). These methods are described in his book Approximation of Large-Scale Dynamical Systems [4]. Orthogonal eigenvalue decomposition is a data analysis method that enables us to propose a reduction of a high-dimensional system by a much lower dimension.

As for Krylov methods, these are iterative methods consisting in projecting the initial problem onto a sequence of Krylov subspaces, transforming the problem into a reduced-size problem that can be solved by direct methods and whose solution is an approximation of the solution of the initial problem.

2.2.5 Conclusion

In this chapter, we have presented the main problems related to reachability and observability for complex control systems in the linear case. It is difficult to predict the behavior of a large, complex system for several reasons. Firstly, complex systems often exhibit emergent properties, i.e. behaviors or characteristics that result from interactions between system components. On the other hand, the topology of large complex systems can sometimes change over time due to internal dynamics and external influences, making it difficult to capture a static snapshot or accurately predict future states. Simulating or modeling large complex systems often requires significant computational power.
We have seen two different approaches. The first is based on numerical algebra methods for the reduction of large scale scale dynamical systems. These simplifications do not require the use of network topology information. The second is based on structural approach and decomposition methods. It makes use of the knowledge on the network topology, but it doesn't take into account quantitative information on the dynamics and does not provide any information on the control cost. It is therefore unable to manage control cost constraints. In this thesis, we propose to use information theoretic tools to solve these problems. We will see that these tools allow us to determine the structure of the system by studying its behaviour without knowing its topology. We will also see that we can obtain means of quantifying the control cost.

2.3 Some basic concepts from information theory

2.3.1 Introduction

The aim of this section is to present some concepts from the theory of information. In 1948, Claude Shannon presented the foundations of this in his article "A Mathematical Theory of Communication" [98]. It is a probabilistic theory that has provided tools for quantifying information, evaluating the performance of information systems. Since the 1950s, information theory has undergone considerable development and has been applied in many fields. In the same period, Wiener also worked on the concept of information, his results were published in 1948 in the book "Cybernetics, or Control and communication in the animal and the machine" [112]. The measures of information proposed by Shannon and Wiener are equivalent. In this chapter, the results concerning entropy and mutual information are given without proof but it is possible to find them in the books "Elements of Information Theory" from Cover [25] and "Théorie de l'information et du codage" from Rioul [93].

2.3.2 Quantity of information

We begin by presenting the measurement of the quantity of information. In this section, we consider $(\Omega, \mathcal{A}, \mathbb{P})$ a probabilized space, Ω is a set named univers, $\mathcal{B}$ is a tribu, and $\mathbb{P}$ is a probability. To quantify the information of an event $A \in \mathcal{A}$, the chosen measure $I(A)$ was constructed to satisfy these 3 requirements:

- If $\mathbb{P}(A) = 1$ then $I(A) = 0$ (learning that a certain event has occurred does not provide information).
- The information decreases as a function of the probability of the event.
- If A and B are independent events then $I(A \cap B) = I(A) + I(B)$

The following definition has been presented to meet these requirements [91]:

Definition 2.3.1. Let A be an event, the information of A is the quantity $I(A)$ defined by

$$I(A) = -\log(\mathbb{P}(A))$$

The choice of the base for the logarithm is arbitrary. We will use the base 2, which is a natural choice when the information is measured in bits.

2.3.3 Entropy

Let $(\Omega, \mathcal{A})$ be a probabilistic space, Ω is a set named univers, $\mathcal{A}$ is a tribu, a set of evenements.
We will now define the entropy of a random variable. This measure is a way of quantifying disorder, i.e. quantifying the unpredictability of the values taken by this random variable.

Definition 2.3.2. Let X be a discrete random variable with a distribution $\big(p(x)\big)_{x\in E}$, where E is the set of values taken by X ($X(\Omega) = E$), the entropy of X is a number $H(X)$ which is the average information of events $\{X = x\}$, i.e. $H(X)$ is defined by

$$H(X) = -\sum_{x\in E} p(x)\log(p(x))$$

with the convention $0 \times \log(0) = 0$

Property 1. Let X be a discrete random variable with a distribution $\big(p(x)\big)_{x\in E}$ and $E = \{x_1, x_2, ..., x_n\}$, we have:

1. $H(X) \geq 0$
2. $H(X) \leq \log(n)$
3. If X is uniformly distributed over $\{x_1, x_2, ..., x_n\}$ then $H(X) = \log(n)$
   In this case, the uncertainty about the value of X is maximal.

Joint entropy, conditional entropy

Definition 2.3.3. Let X and Y be two discrete random variables with joint distribution $p(x, y)$. The joint entropy of X and Y is defined as the entropy of the pair (X, Y):

$$H(X,Y) = -\sum_{x}\sum_{y} p(x,y)\log(p(x,y)) \tag{2.3.1}$$

The definition of a pair of random variables will enable us to define the conditional entropy. We saw above the definition of the entropy of a random variable, which measures the uncertainty of the values taken by this random variable. The uncertainty of a random variable Y knowing the value of another random variable X may be measured by the conditional entropy.

Definition 2.3.4. Consider two discrete random variables X and Y with joint distribution $p(x, y)$, the conditional entropy of Y knowing X is defined as

$$H(Y|X) = -\sum_{x} p(x)H(Y|X = x) \tag{2.3.2}$$

$$H(Y|X) = -\sum_x p(x)H(Y|X=x) \tag{2.3.3}$$

$$= -\sum_x p(x)\sum_y p(y|x)\log p(y|x) \tag{2.3.4}$$

$$= -\sum_x \sum_y p(x)p(y|x)\log p(y|x) \tag{2.3.5}$$

$$= -\sum_x \sum_y p(x,y)\log(p(y|x)) \tag{2.3.6}$$

We have the following bound of $H(X|Y)$.

Theorem 2.3.5. Let X and Y denote two discrete random variables. We have:

$$H(X|Y) \leqslant H(X)$$

with equality if and only if X and Y are independent.

The definition of the joint entropy can be generalized with n random variables $X_1, ..., X_n$,

Definition 2.3.6. Let be $(X_1, X_2, ..., X_n)$ n discrete random variables with joint distribution $p(x_1, x_2, ..., x_n)$. The joint entropy of $(X_1, X_2, ..., X_n)$ is defined as the entropy of the vector $(X_1, X_2, ..., X_n)$:

$$H(X_1, X_2, ..., X_n) = -\sum_{x_1, x_2, ..., x_n} p(x_1, x_2, ...x_n)\log(p(x_1, x_2, ...x_n)) \tag{2.3.7}$$

Generalizing the conditional entropy, we obtain these following theorem:

Theorem 2.3.7. (Chain rules for entropy) Let $X_1, X_2, ..., X_n$ n random variables then

$$H(X_1, X_2, ..., X_n) = \sum_{i=1}^{n} H(X_i|X_{i-1}, ..., X_1) \tag{2.3.8}$$

$$H(X_1, X_2, ..., X_n) \leqslant \sum_{i=1}^{n} H(X_i) \tag{2.3.9}$$

with equality if and only if X_i is independent of $(X_{i-1}, X_{i-2}, ...X_1)$ for all i (i.e. if and only if the X_i are independant).

2.3.4 Mutual information

To measure the dependance between two random variables X and Y the idea is to compare the entropy $H(X)$ and the conditional entropy $H(X|Y)$, or by symmetry, $H(Y)$ and $H(Y|X)$. Therefore, we are interested in the values $H(X) - H(X|Y)$ and

$H(Y) - H(Y|X)$. Using 2.3.2 and 2.3.4, we obtain that these values are equal. This quantity is called mutual information. It can also be defined as the Kullback-Leibler divergence, i.e. the "distance" between the distribution $p(x, y)$ and $p(x) \times p(y)$ as follow.

Definition 2.3.8. Let X and Y be two discrete random variables with joint distribution $p(x, y)$ and marginals $p(x)$ and $p(y)$. The mutual information between X and Y is defined as :

$$I(X, Y) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{2.3.10}$$

Relationships between mutual information, entropy, joint entropy and conditional entropy, are collected in the following theorem and represented in figure (2.11).

Theorem 2.3.9.

$$H(X;Y) = H(X) + H(Y|X) \tag{2.3.11}$$
$$= H(Y) + H(X|Y) \tag{2.3.12}$$
$$I(X;Y) = H(X) - H(X|Y) \tag{2.3.13}$$
$$I(X;Y) = H(Y) - H(Y|X) \tag{2.3.14}$$
$$I(X;Y) = H(X) + H(Y) - H(X;Y) \tag{2.3.15}$$

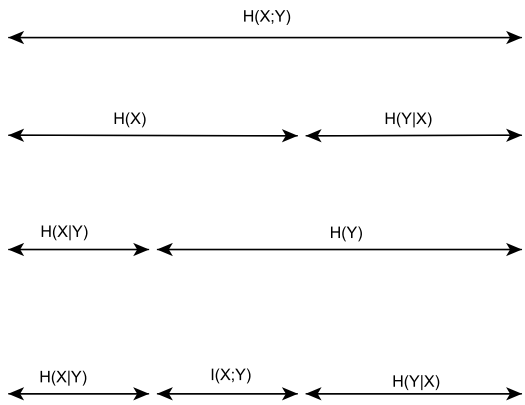

Figure 2.11 – Relationships between mutual information, entropy, joint entropy and conditional entropy

In the case where we consider the mutual information between a random variable Y and a vector $(X_1, X_2, ..., X_n)$, we have a chain rule for information:

Property 2.

$$I((X_1, ..., X_n); Y) = \sum_{i=1}^{n} I(X_i; Y|(X_{i-1}, ..., X_1)) \tag{2.3.16}$$

In complex systems, multi-information makes it possible to study the influence of an agent's state on the overall state of the system. Later on, we will use this notion to bring the system to a desired state at minimum cost.

2.3.5 Capacity of a channel

A communication chain (see Fig (2.12)) includes a discrete source. This source generates a message constituted by a sequence of symbols $\{x_1, x_2, ..., x_n\}$, each of them belonging to an alphabet. The source is characterized by the random variable X. When a message goes through such a channel, it's routed between its input and its output. The last element of the chain of transmission is the receiver. The random variable Y characterizes the received message constituted by a sequence of symbols $\{y_1, y_2, ..., y_p\}$. This alphabet may be different from the alphabet of the source message.

The channel itself is represented by transition matrix A which allows us to obtain the output distribution knowing the input distribution :

$$A = \begin{pmatrix} p(y_1|x_1) & p(y_1|x_2) & \dots & p(y_1|x_n) \\ p(y_2|x_1) & \dots & \dots & p(y_2|x_n) \\ \vdots & & & \vdots \\ p(y_p|x_1) & \dots & \dots & p(y_p|x_n) \end{pmatrix}$$

The capacity of the channel is the maximum of the mutual information between the input and output messages, where the maximization is with respect to the input distribution. We have the following definition :

Figure 2.12 – A communication chain

Definition 2.3.10. The capacity of the channel is the real C defined by

$$C = \max_{P_X} I(X, Y) \tag{2.3.17}$$

where P_X is the probability distribution of the source

The capacity of a channel is the upper limit of the information rate that can be reliably transmitted over that communication channel. This corresponds to the maximum information data rate that can pass through the channel.
In the case of a symmetric binary channel with an error probability ϵ, the transition matrix is

$$A = \begin{pmatrix} 1-\epsilon & \epsilon \\ \epsilon & 1-\epsilon \end{pmatrix}$$

We note $\begin{pmatrix} p(x_0) \\ p(x_1) \end{pmatrix} = \begin{pmatrix} P(X=0) \\ P(X=1) \end{pmatrix}$ and $\begin{pmatrix} p(y_0) \\ p(y_1) \end{pmatrix} = \begin{pmatrix} P(Y=0) \\ P(Y=1) \end{pmatrix}$.
We take any input probability distribution $\begin{pmatrix} p(x_0) \\ p(x_1) \end{pmatrix} = \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix}$
We have

$$\begin{pmatrix} p(y_0) \\ p(y_1) \end{pmatrix} = A \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix} = \begin{pmatrix} \epsilon + (1-2\alpha)\epsilon \\ (1-\epsilon) - (1-2\alpha)\epsilon \end{pmatrix} \tag{2.3.18}$$

Thus

$$H(Y) = H_2(\epsilon + (1-2\epsilon)\alpha) \tag{2.3.19}$$

On the other hand

$$H(Y|X) = -\sum_{i=0}^{1}\sum_{j=0}^{1} p(y_j|x_i)p(x_i)\log(p(y_j|x_i)) \tag{2.3.20}$$

$$= -(1-\epsilon)\alpha\log(1-\epsilon) - \epsilon(1-\alpha)\log(1-\epsilon) \tag{2.3.21}$$

$$- \epsilon\alpha\log(\epsilon) - (1-\epsilon)(1-\alpha)\log(1-\epsilon) \tag{2.3.22}$$

$$= -\epsilon\log(\epsilon) - (1-\epsilon)\log(1-\epsilon) \tag{2.3.23}$$

$$= H_2(\epsilon) \tag{2.3.24}$$

According to the theorem 2.3.15 and the equation above, we have:

$$I(X,Y) = H(Y) - H(Y|X) \tag{2.3.25}$$

$$= H_2(\epsilon + (1-2\epsilon)\alpha) - H_2(\epsilon) \tag{2.3.26}$$

As the capacity of the channel is $C = \max_{\alpha\in[0;1]} I(X,Y)$,
we have $C = max_{\alpha\in[0;1]}\big(H_2(\epsilon + (1-2\epsilon)\alpha)\big) - H_2(\epsilon)$
For $\alpha = \frac{1}{2}$, we have $I(X,Y) = H_2(\frac{1}{2}) - H_2(\epsilon)$

The channel capacity is therefore:

$$C = 1 - H_2(\epsilon) \tag{2.3.27}$$

For a symmetric binary channel, when we have $C = 1$ this means that there are no transmission errors; when $C = 0$, the output data cannot be used to deduce the input

data.
In information theory, capacity is a value that allows us to quantify the loss of information in messages passing through a transmission channel. It is therefore a very important tool when choosing the construction of corrector codes for transmitting information. In the study of controllability problems, this notion of capacity can be very useful. It allows us to quantify our level of control of a system by forcing an agent. As capacity corresponds to the maximum threshold of information that can pass through an information channel between 2 agents, by forcing the state of one agent, we know how we can influence the states of the other agents.

In the next chapter, we will present the link between information theory and the controllability of systems. First, we will define the delayed mutual information, which will allow us to detect causal links between the states of different agents in a system.

3 | Observability of a complex system: link with information theory

3.1 Introduction and review

3.1.1 The concept of Causality

The concept of causality and the characterization of influence are very important in many areas of science [84]. In the theory of control, it is fundamental to know which variable to modify in a complex system in order to obtain a desired state. It is necessary to identify the causal structure of a dynamical complex system in order to better understand its dynamical behavior. In particular, it reveals how the different degrees of freedom of a system influence each other. Once we have the causal model of the system, it allows us to represent qualitative information and evaluate the influence of a relevant variable on the overall performance of the system. Such a qualitative representation format was developed by Pearl [85] in the context of the theory of causal models.
The notion of causality has been raised by philosophers since antiquity. It's still difficult to define, and it is important not to confuse causality with correllation. Two events can have common causes, without there being a causal link between them. In many scientific fields, the theory of probabilities is used to characterize causality.
Suppes [105] did describe in 1970 the probabilistic notion of causality as follows: An event A is a cause of event B if :

- A occurs before B
- the probability of A is non-zero
- the probability of B knowing A is greater than the probability of B

This definition is unsatisfactory because, with probabilities, we can detect the correlation between events but not necessarily the causality.
Granger [40], [41], based on work by Wiener [111] in 1956, was the first to propose a practical, operational definition of causality based on prediction improvement. The

concept of Granger causality is a characterization of causality between sequences of data with a time lag. This characterization has been widely used in economics since the 1960s. In recent years, Granger's concept of causality has become very popular in neuroscience, as a way of determining the causal links between different brain areas. [97], [104].
In this thesis, our objective in causal modeling is not only to determine a cause and effect relationship, but also to measure or quantify the intensity of this relationship. Information theory offers tools to evaluate the intensity of this causality.

3.1.2 Controllability and information theory

Schreiber [96] defined a measure using the notion of mutual information which he named transfer entropy, and which took into account the time lag between variables variations. This tool allows us to determine the Granger Causality and has been used in many fields. For example, in neurosciences, Amblard et al [2], [3] show that transfer entropy can be used to assess Granger causality graphs of stochastic processes. Becq and Amblard use it to detect Granger causality graphs in the analysis of electroencephalographic signals [7] .

3.1.3 Measure of the influence: Delayed Mutual Information

We used a tool similar to transfer entropy that we named delayed mutual information, inspired by the work of Gregor Chliamovitch [20].

Let us consider a set of random variables $X_i(t)$ associated with agent i of a complex system, taking values in a set E. For instance, $X_i(t) = s_i(t)$ means that the state of agent i at time t takes the particular value $s_i(t) \in E$
To measure the influence between agents i and j, we define the τ-delayed mutual information $w_{i,j}$: it's the mutual information (see the definition 2.3.8) between the random variables $X_i(t)$ and $X_j(t+\tau)$.

Definition 3.1.1. The τ-delayed mutual information between agents i and j at time t is:

$$\begin{aligned} w_{i,j}(t,\tau) &= I(X_i(t), X_j(t+\tau)) && (3.1.1)\\ &= \sum_{(x,y)\in E^2} p_{xy} \log\left(\frac{p_{xy}}{p_x p_y}\right) && (3.1.2)\end{aligned}$$

with

$$p_{xy} = \mathbb{P}(X_i(t) = x, X_j(t+\tau) = y)$$

$$p_x = \mathbb{P}(X_i(t) = x) \text{ and } p_y = \mathbb{P}(X_j(t+\tau) = y)$$

We notice that delayed mutual information is not a reflexive value (generally $w_{i,j}(t,\tau) \neq w_{j,i}(t,\tau)$).

We will use the delayed mutual information $w_{i,j}(t,\tau)$ to quantify the influence of the value of X_i at time t on the value of X_j at time $t+\tau$. To know the influence of X_i on the system shifted by time τ, we need to determine the mutual information between X_i and a measure which globally characterizes the state of the system.

3.1.4 Measure of the influence of one agent on the whole system: Delayed Multi-Information

Let $Y(t)$ be a random variable which characterizes the state of the system at the time t. We also define the τ-delayed multi-information w_i to measure the influence of the agent i on the system.

Definition 3.1.2. The τ-delayed multi-information of an agent i at the time t is:

$$w_i(t,\tau) \quad = \quad I(X_i(t), Y(t+\tau)) \tag{3.1.3}$$

We can take for instance:

$$Y(t+\tau) = \sum_k X_k(t+\tau) \tag{3.1.4}$$

When the values of the random variable X_i are Boolean, Y in 3.1.4 gives the number of agents with the property $X(t+\tau) = 1$. In that case, the multi-information allows us to identify the most influential agents on the system, and possibility those agents to force in order to control the overall system.

3.1.5 Calculation of the mutual information

To compute the mutual information, it is necessary to know the probability distributions of the 2 random variables and the joint law: the probability distributions of $X_i(t)$ and $X_j(t+\tau)$ (resp. $X_i(t)$ and $Y(t+\tau)$) together with their joint distribution for the delayed mutual information (resp. the delayed multi-information). If these distributions are unknown, it is possible to obtain them by sampling, either in simulation or by observing the behavior of the system. When the samples obtained are sequences of independent random variables with the same law which admits a moment of order 2, we can use the central limit theorem. It gives us the accuracy of these probability estimations as a function of the sample size. In the sequel, we will consider set averages of values obtained in many simulations. Since these simulations will be independent, the sample values will be independent as well.

If we don't know the dynamics of the system or if these dynamics don't allow the calculation of these distributions, we can use the central limit theorem to obtain approximate values of these probability laws. We observe the behavior of the system, and we note at each simulation the values of the state at time t and $t + \tau$. We thus obtain a sample of values which allows us to know approximate values of the probabilities of the law of the random variable considered. We can evaluate the precision of these values in function of N, the size of the sample.

We consider an attribute of the members of a population which appears with probability p. For a sample of size N drawn in this population, let F_n be the random variable equal to the proportion of those elements having this attribute. According to the Moivre-Laplace theorem, the random variable $\frac{F_N - p}{\sqrt{p(1-p)/N}}$ converges in distribution to a Gaussian distribution: $\frac{F_N - p}{\sqrt{p(1-p)/N}} \underset{n \to +\infty}{\longrightarrow} Y$.

Therefore we may choose the size N of the sample according to the desired accurracy and risk for the values of the marginal and joint distributions. To compute the τ-delayed mutual information , we then run N simulations between the instants t and $t + \tau$, we obtain the states $S(i, \delta)$ value of the random variable X_i at the instant δ. We initialize to 0 the matrix $N^{p,q}$, and we apply algorithm 1.

Algorithm 1 Compute τ-delayed Mutual Information

M the model

N is the number of simulations.

n is the number of agents.

SIMULATION is a procedure that executes a simulation and returns the list of agent states at each time.

procedure IM=MUTUAL(M, N)
  for k from 1 to p **do**
    for ℓ from 1 to q **do**
      $N^{k,\ell} \leftarrow 0_{n,n}$ ▷ The coefficients of $N^{k,\ell}$ are the number of time where we have the couple of state (k, ℓ)
    end for
  end for
  for p from 1 to N **do** ▷ Run the procedure which execute simulations
    $S \leftarrow SIMULATION(Model, t, t+\tau)$ ▷ S is the array of states at the instant t and $t+\tau$
    for i from 1 to n **do**
      for j from 1 to n **do**
        $k \leftarrow S(i,t)$
        $\ell \leftarrow S(j,t)$
        $N^{k,\ell}(i,j) = N^{k,\ell}(i,j) + 1$
      end for
    end for
  end for
  for i from 1 to n **do**
    for j from 1 to n **do**
      $$IM(i,j) \leftarrow \sum_{k=0}^{p}\sum_{\ell=0}^{q} \frac{N^{k,\ell}(i,j)}{N} \log\Big(\frac{N \times N^{k,\ell}(i,j)}{\sum_{r=0}^{p} N^{r,\ell}(i,j)}\Big)$$
    end for
  end for
  Return IM
end procedure

The complexity of this algorithm is $O(N \times t \times n^2)$. The coefficients i, j of $N^{p,q}$ are the number of times the states of agent pair of agents i and j are equal to (p, q) at respective times t and $t+\tau$.

3.2 The voter model

3.2.1 Introduction

In this chapter, we present the model we have used to verify that information theory offers tools that can be effective in enabling us to approach solutions to the problems of controllability and observability.
We chose the voter model because it is a fairly simple model but with enough richness to be not trivial.
Moreover, it offers a variety of behaviors, it applies to all types of states of a system and may be applied for all topologies.
It is also close to other models, such as the epidemic models or neural circuits for instance.
Castellano et al [18] defines a q-voter model in which an agent votes like its neighbors if the opinion is unanimous; otherwise the vote is random. This model is used by Nyczka et al. [78],[79]. In their paper, They study how different types of nonconformity, introduced at the microscopic level, manifest themselves at the corporate level. Our model is closer to those by Mobilia et al. [70] or Masuda [63] because, in these models, agent's vote depends on his neighbor's majority vote. Brede et al. [15] and Kuhlman et al. [52] propose a model where an agent randomly chooses one of its neighbors, and decides to vote like him the next time. With this dynamics, Kuhlman et al. study the effect of controlling the vote of some agents on the system for different graphs. In the literature, there are many articles that deal with the effects of zealots, an agent that does not change its vote, as defined by Mauro Mobilia [69], or inflexible as defined by Galam and Jacobs [36]. For instance, in [70],[69] Mobilia et al. study the role of zealots on the result of a vote and Masuda [63] showed the link between the role of zealots and their degree (the higher the zealot's degree, the more influence). Morone et al. [71] have propose a numerical method to find these zealots by using the adjacency matrix of the graph.
We will now describe the dynamics of the voter model used for the simulations in the next section.

3.2.2 Description of the Voter Model

In the voter model that we have chosen for our investigations, a binary agent occupies each node of a network. The dynamics is specified by assuming that each agent i looks at every other agent in its neighborhood, and counts the percentage ρ_i of those which are in the state $+1$ (in case an agent is linked to itself, it belongs to its own neighborhood). A function f is specified such that $0 \leq f(\rho_i) \leq 1$ and gives the probability for agent i to be in state $+1$ at the next iteration. For instance, if f is

chosen as $f(\rho) = \rho$, an agent for which all neighbors are in state $+1$ will turn into state $+1$ with certainty. The update is performed synchronously over all n agents. Formally, the dynamics of the voter model can be express as

$$s_i(t+1) = \begin{cases} 1 & \text{with probability } f(\rho_i(t)) \\ 0 & \text{with probability } 1 - f(\rho_i(t)) \end{cases} \tag{3.2.1}$$

where $s_i(t) \in \{0, 1\}$ is the state of agent i at iteration t, and

$$\rho_i(t) = \frac{1}{|N_i|} \sum_{j \in N_i} s_j(t). \tag{3.2.2}$$

The set N_i is the set of agents j that are neighbors of agent i, as specified by the network topology. The global density of all n agents with opinion 1 is obtained as

$$\rho(t) = \frac{1}{n} \sum_{i=1}^{n} s_i(t) \tag{3.2.3}$$

The dynamics of the system may also depends on the noise value ϵ which is the probability that the agent takes a decision different from the majority of its neighborhood. In what follows, we will use a particular function f: (see Fig. 3.1)

$$f(\rho, \epsilon) = (1 - \epsilon)\rho + \epsilon(1 - \rho) = (1 - 2\epsilon)\rho + \epsilon \tag{3.2.4}$$

Let $Y_i(t)$ be the random variable equal to 1 when agent i's neighbors vote majority 1 and 0 otherwise at time t.

According to the law of total probability, the probability p_i that agent i votes $+1$ is

$$\begin{aligned} p_i(t+1) &= \mathbb{P}(X_i(t+1) = 1) \\ &= \mathbb{P}_{Y_i(t)=1}(X_i(t+1) = 1)\mathbb{P}(Y_i(t) = 1) + \mathbb{P}_{Y_i(t)=0}(X_i(t+1) = 1)\mathbb{P}(Y_i(t) = 0) \\ &= (1 - \epsilon)p_{V_i}(t) + \epsilon(1 - p_{V_i}(t)) \\ &= (1 - 2\epsilon)p_{V_i}(t) + \epsilon \end{aligned}$$

where $p_{V_i}(t)$ is the probability that the majority of neighbors of agent i vote 1 at time t ($p_{V_i(t)} = \mathbb{P}(Y_i(t) = 1)$).
We limit the noise ϵ in the range $0 \leq \epsilon \leq 1/2$. The upper value $\epsilon = 1/2$ corresponds to a blind vote, with probability $1/2$ for each outcome.

3.2.3 Representation of the social network

To illustrate the behavior of this model, we consider a scale-free graph G [6], to model a social network, as proposed by Newman et al. in the paper Random graph models

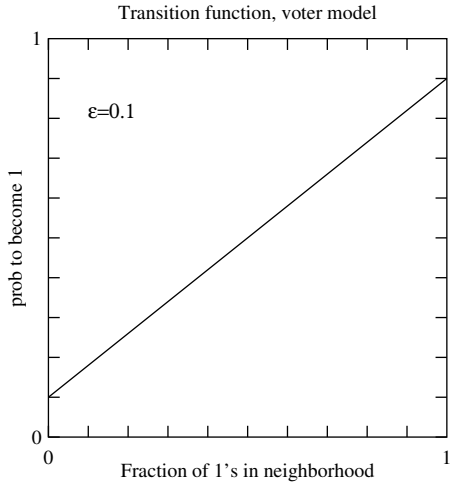

Figure 3.1 – The probablility $f(\rho)$ used in this study. The noise ϵ is visible as the values of $f(0)$ and $1 - f(1)$.

of social networks [77]. In a scale-free network, a small number of nodes have many connections. In a social network, these nodes are the most popular people, the ones with the highest number of social links, they are the hubs of the graph. The nodes typically have very few connections. The majority of voters are represented by these nodes. The scale free graph structure is based on communities built around a hub (see the paper by Wu et al. [116]). We use the algorithm from Barabási-Albert detailed in the paper Bollobás et al. [13], to generate a random scale-free graph. In figure 3.2, we can see a graph of order 100 generated with this algorithm.

3.2.4 Simulation

Simulations for the voter model are performed using algorithm 2 presented here after. The input values are the adjacency matrix, the noise and the duration of the simulation. The output value is the state matrix $S = (s_{i,j})$ with $s_{i,j}$ the state of the agent i at the time j. Figure 3.3 shows the corresponding density of agents with opinion 1, as a function of time. We can see that there is a lot of fluctuations due to the fact that states "all 0's" or "all 1's" are no longer absorbing states when $\epsilon \neq 0$. As can be seen on these figures, the amplitudes of the fluctuations are smaller for larger graph sizes (regularization of the average density of agents with opinion 1 with the size of the sampling).

Algorithm 2 Simulation of the Voter Model

Input variables are:

A the adjacency matrix of the graph that represent the system.

ϵ the noise.

t the duration (time interval) for the simulation..

Output variable is:

S the list of states of each agent at each time.

procedure S=SIMULATION(A, ϵ, t)

  $n \leftarrow size(A, 1)$ ▷ n the number of rows in A (number of agents)

  $deg \leftarrow A \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ ▷ agents degree vector

  $S \leftarrow 0_{n,t}$ ▷ S is the matrix of states

  */ we randomly initialize the vote of each agent at time 1

  for k from 1 to n **do**

    $S(i, 1) = Random(0, 1) < 0.5$ ▷ Random(0,1): uniform law on $[0; 1]$

  end for

  for j from 2 to t **do**

    $V \leftarrow A \times S(:, j-1)$ ▷ vector V: number of neighbors who vote 1

    for i from 1 to n **do**

      $\rho \leftarrow V(i, j)/deg(i)$

      $S(i, j) = Random(0, 1) < (1 - 2\epsilon)\rho + \epsilon$

    end for

  end for

  Return S

end procedure

Figure 3.2 – A scale-free graph of order 100. Blue vertices have a degree greater than 10, green vertices have a degree between 7 and 10

3.3 Results with the voter model

3.3.1 Characterization of the influence of an agent

In this study we would like to characterize how the opinion of one agent influences that of its neighbors and that of the entire system. We will first propose an approach based on information theory, and then measure the influence directly by forcing (or controlling) the opinion of one agent. We will show that both characterizations are strongly correlated. The information theoretic quantities that will be considered are the time-delayed mutual information and the time-delayed multi-information. The idea is to consider a time delay large enough to capture the causal effect of one element on another.

First, we present the results obtained for detecting the influential agents of the system for a voter model where the social network is defined by a scale-free graph. Next, we consider a 1D voter model that allows us to define a value called the control length,

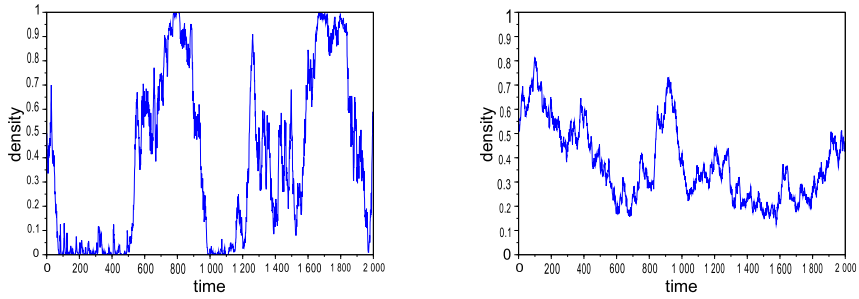

Figure 3.3 – Plot of the time evolution of the density of opinion 1 with noise $\epsilon = 0.001$ and $n = 200$ agents (left) or $n = 2000$ agents (right).

which enables us to evaluate an agent's radius of influence. Finally, in the case of this 1D voter model, we relate the results obtained to the general results of control theory presented in chapter 2.2.

Delayed mutual- and multi-information

Let us consider a set of random variables $X_i(t)$ associated with each agent i, taking their values in a set A. For instance, $X_i(t) = s_i(t)$ would be the opinion of agent i at iteration t.
To measure the influence between agents i and j, we use the τ-delayed mutual information $w_{i,j}$ (see definition 2.3.8). As explained in Chapter 2, the delayed mutual information $w_{i,j}(t, \tau)$ quantifies the influence of the vote of agent i at time t on the vote of the agent j at time $t + \tau$.

We also use the τ-delayed multi-information $w_i(t, \tau)$ to measure the influence of the vote of agent i at instant t on the sum of the vote of the others at instant $t + \tau$ (see definition 3.1.3).

Non-intrusive characterisation of the node influence: the delayed multi-information

We wish to calculate the multi-information to identify the most influential agents in the system. We take a scale free graph of order 50 to represent a social network (see figure 3.4). Using the voting dynamics defined in subsection 3.2.2, we calculate the multi-information by sampling, as described in subsection 3.1.5. Figure. 3.5 shows $w_i(100, 2)$ in a stationary regime, where the origin of time is arbitrary. We observe that some agents exhibit a more pronounced peak of multi-information towards the

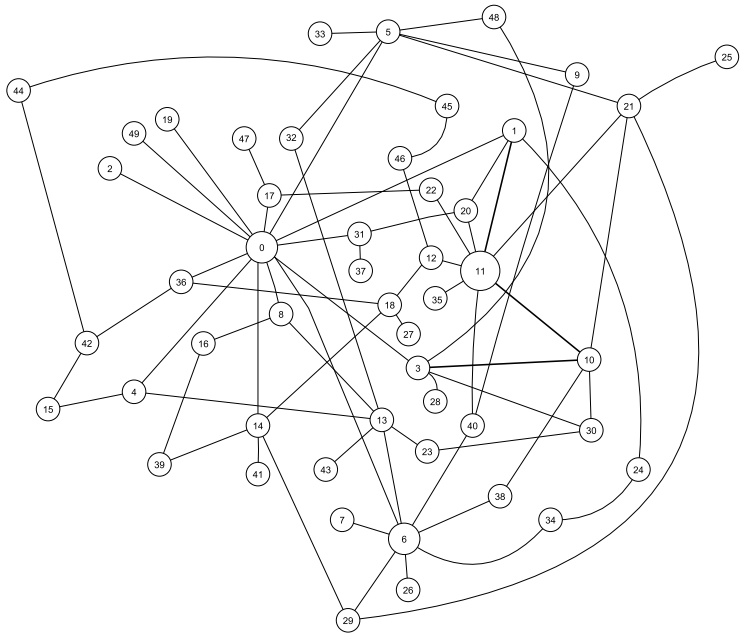

Figure 3.4 – Graph G of the social network from the example in subsection 3.3.1

rest of the system, suggesting that the opinion of these agents may affect more the global opinion of all agents. Note that this results is obtained only by probing the systems, without modifying any of its components. For this reason, we describe this approach as "non-intrusive". The algorithms used throughout this section to numerically evaluate the delayed mutual- and multi-informations in the voter model example are described in the section 3.1.5.

As we can see in Figure 3.6, the influence of each agent decreases strongly as the noise increases. We deduce that with a high noise it will be difficult to control the system. Brede et al. [15] obtained similar results about the effect of the noise on the control.

Intrusive characterization: forcing

In this subsection, we consider another way to measure the influence of an agent on the system. We call this approach "intrusive" as it implies a perturbation, and is no longer just an observation.

To measure the influence of agent i, its opinion is forced to a chosen value, for instance

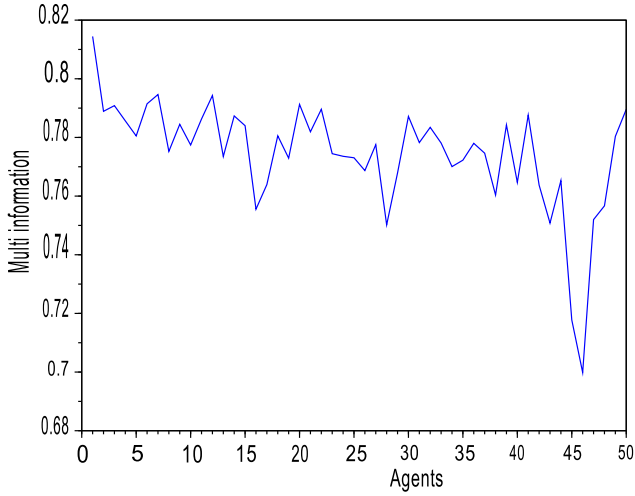

Figure 3.5 – τ-delayed multi-information $w_i(100, 2)$ as a function of i, for graph G with $n = 50$ agents and noise level $\epsilon = 0.001$.

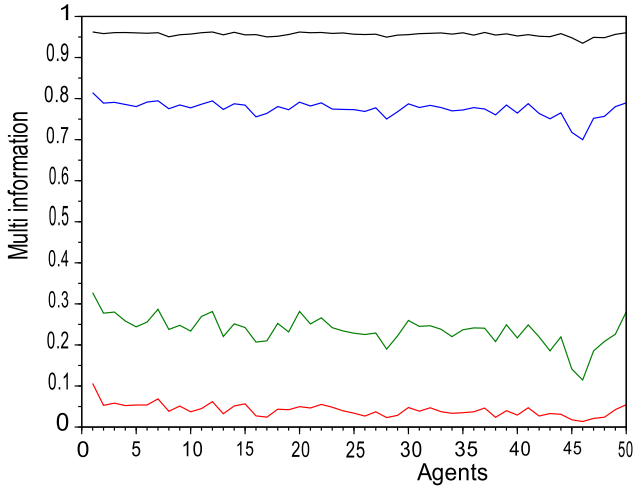

Figure 3.6 – τ-delayed multi-information $w_i(100, 2)$ as a function of i, for graph G of the figure 3.2 with $n = 50$ agents and with different noise level $\epsilon = 0.0001$ (black curve), $\epsilon = 0.001$ (blue curve), $\epsilon = 0.01$ (green curve) and $\epsilon = 0.05$ (red curve).

the value 1. As a result the density (3.2.3) of opinion 1 in the system

$$\rho_i(t) = \frac{1}{n}\sum_{j=1}^{n} s_j(t) \tag{3.3.1}$$

can be averaged over a large number N of independent realizations, to give a quantity $\langle\rho_i(t)\rangle$, where the subscript i indicate which agent has been forced to 1. If t is large enough, $\langle\rho_i\rangle$ no longer depends on t.
The influence can be measured in stationary regime, or from the initial state where all agents are initialized uniformly to 0 or 1 with probability 1/2, respectively.
The color representations of the graphs (Figures 3.7 and 3.8) show that the multi-information gives some information about the controllability of the system. In the case the multi-information is calculated from the initial state, these figures emphasize the link between the multi-information and the influence of an agent. We can then identify the agents that allow the best control of the system when their vote is forced. With the multi-information, we get the same results as Masuda [63]: choosing hubs as zealots is a good strategy to control the system and the effect of the control is strongly correlated with the hub's degree.
The measurement obtained in the steady state for the delayed multi-information is different from that observed in the transient regime. Low-impact agents can get a high multi-information by being a proxy of an influential neighbor. In this case, the multi-information evaluates the observability rather than the controllability, i.e. the state of this agent with little influence informs us about the state of the system, which is strongly influenced by its neighbor, which is a hub.

3.3.2 The 1D voter model

The previous section gave an illustration of the link between influence defined by intrusive forcing and the influence measured by observing the time-delayed multi-information. In this section, we propose an analytical mean field solution of the voter model, in a one-dimensional topology. This solution will formally specify the previously observed link. In particular we will introduce a characteristic control length.

Presentation

We consider the case of n voters organized along a line so that voter i looks at voter $i-1$ and itself to take its decision. This system can be modeled by the directed graph shown in figure 3.9.

Chain network have been already studied. Brede et al. [16] has demonstrated that the influence of indirect control on an agent decreases with the distance from the

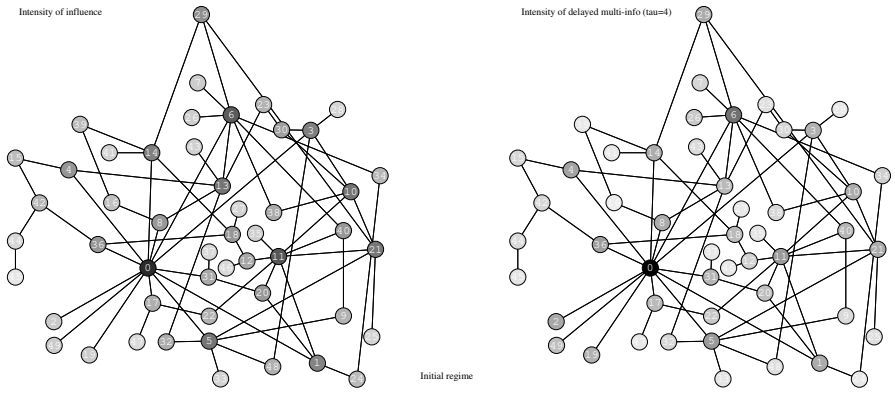

Figure 3.7 – Scale free graph colored as a function of the values of the influence (left) and the τ-delay multi-information (right), for $\tau = 4$. In this case, the multi-information is computed during the transient dynamics.

controlled agent when there is noise. Let's take a closer look at the link between influence and distance.

Agent $i = 0$ has no left neighbor and will have a controlled dynamics. For instance its opinion will be always forced to 1. The other agents are initialized randomly in $\{0, 1\}$.

Since agent 1 is looking at agent 0, its next state will likely to be 1. And so on for agent $2, 3, \ldots, n$. Intuitively, we could expect that the entire system will become 1, due to the control imposed by agent 1. But noise is changing this conclusion.

If $p_i(t)$ is the probability that agent i is 1 at time t, we can write the equation

$$p_i(t+1) = p_i(t)W_{1\to1}(t) + (1 - p_i(t))W_{0\to1}(t) \tag{3.3.2}$$

where $W_{a\to b}$ is the probability that the state evolves from a to b at the next time step. The probability for an agent to vote 1 is $f(\rho) = (1 - 2\epsilon)\rho + \epsilon$ as we see in the subsection 3.2.2. In the case 1D, for the agent i, ρ can only take the value 0 (when $s_{i-1}(t) = s_i(t) = 0$), $1/2$ (when $s_{i-1}(t) = 1$ and $s_i(t) = 0$ or when $s_{i-1}(t) + 0$ and $s_i(t) = 1$) and 1 (when $s_{i-1}(t) = s_i(t) = 1$), Hence,

$$\begin{aligned} W_{0\to1} &= p_{i-1}(t)f(1/2) + (1 - p_{i-1}(t))f(0) \\ W_{1\to1} &= p_{i-1}(t)f(1) + (1 - p_{i-1}(t))f(1/2) \end{aligned} \tag{3.3.3}$$

Before attempting to solve the above system analytically, we can observe its behavior numerically. We can see in figure 3.10 that without noise ($\epsilon = 0$), the entire system is

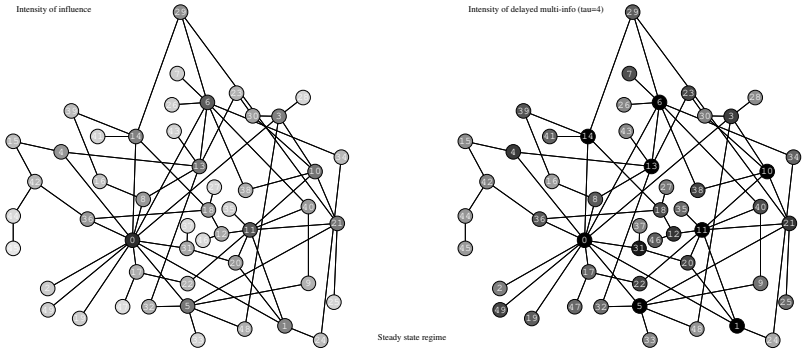

Figure 3.8 – Scale free graph colored as a function of the values of the influence (left) and the τ-delay multi-information (right), for $\tau = 4$. In this case, the multi-information is computed when the system is in a steady state regime.

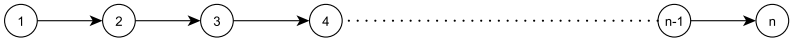

Figure 3.9 – Graph of the 1D voter model. This directed graph indicates that an agent's vote depends on his own vote and on the vote of the one before him in the line.

indeed controlled by the left-most agent whose state is always 1. But, as soon as the noise is increased ($\epsilon = 0.01$) the control is not effective anymore. There is a critical noise $\epsilon = \epsilon_c(n)$ below which a system of size n can be controlled by the first node, and above which the influence of the driving node is diluted by the noise.

Figure 3.11 shows the density of agents with opinion 1, as a function of time, for different intensities of noise, ϵ. We observe in this figure the effect of the system size. For smaller systems, the effect of controlling agent $i = 1$ is more effective than for larger n.

Probability distribution of opinions

We can determine the probability distribution in the case of the linear voter model. From 3.3.2 and 3.3.3. We have

$$\begin{aligned} p_i(t+1) &= p_i(t)W_{1\to1}(t) + (1-p_i(t))W_{0\to1}(t) \\ &= p_i(t)(W_{1\to1}(t) - W_{0\to1}(t)) + W_{0\to1}(t) \end{aligned}$$

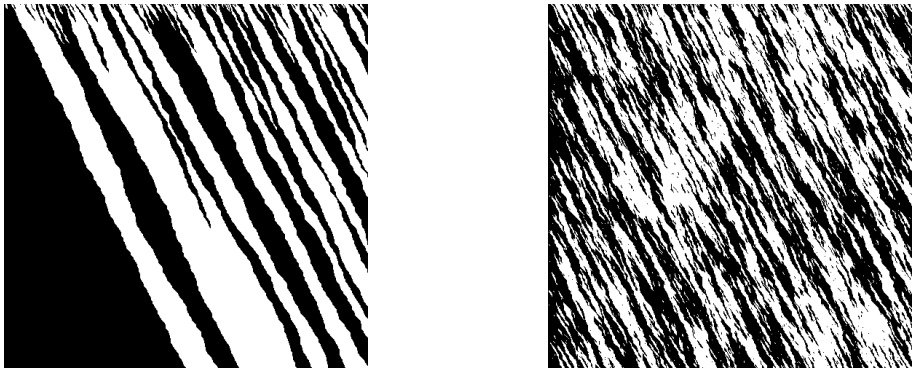

Figure 3.10 – Space-time diagram for the evolution of the state of a linear 1D voter model with $n = 500$ agents. Line t of the figure depicts the configuration of the n agents'votes at iteration t. We can see the first 500 iterations. Left: $\epsilon = 0$. Right: $\epsilon = 0.01$.

with

$$\begin{aligned} W_{1\to1}(t) &= p_{i-1}(t)f(1) + (1 - p_{i-1}(t))f(1/2) \\ &= p_{i-1}(t)(1-\epsilon) + (1 - p_{i-1}(t))\frac{1}{2} \end{aligned}$$

and

$$\begin{aligned} W_{0\to1}(t) &= p_{i-1}(t)f(1/2) + (1 - p_{i-1}(t))f(0) \\ &= p_{i-1}(t)\frac{1}{2} + (1 - p_{i-1}(t))\epsilon \end{aligned}$$

Therefore,

$$p_i(t+1) = \left(\frac{1}{2} - \epsilon\right) p_i(t) + \left(\frac{1}{2} - \epsilon\right) p_{i-1}(t) + \epsilon \tag{3.3.4}$$

As $p_0(t) = 1$, we obtain

$$p_1(t+1) = \left(\frac{1}{2} - \epsilon\right) p_1(t) + \frac{1}{2} \tag{3.3.5}$$

Let $P(t)$ be the vector of probabilities

$$P(t) = \begin{pmatrix} p_1(t) \\ p_2(t) \\ \vdots \\ p_n(t) \end{pmatrix}$$

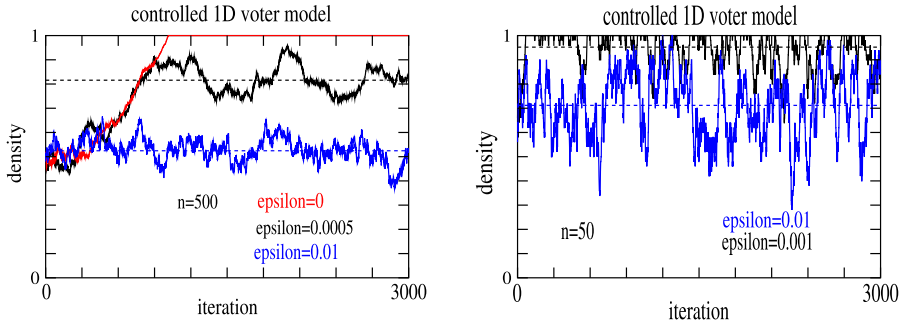

Figure 3.11 – Density of agents with value 1 as a function of time, for different noise levels, and two different system sizes, $n = 500$ and $n = 50$. The dashed lines are the predictions of the mean field analytical approach, see eq. (3.3.17).

3.3.4 can be expressed in a matrix form

$$P(t+1) = AP(t) + B \tag{3.3.6}$$

with

$$A = (\frac{1}{2} - \epsilon) \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ 1 & 1 & 0 & \ddots & \vdots \\ 0 & 1 & 1 & \ddots & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & 1 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 1/2 \\ \epsilon \\ \vdots \\ \epsilon \end{pmatrix}$$

This equation can be solved recursively:

$$P(t) = A^t P(0) + \big(\sum_{j=0}^{t-1} A^j\big) B \tag{3.3.7}$$

Now, we compute power matrices A^j. We have $A = I_n + C$ with

$$C = \begin{pmatrix} 0 & 0 & \dots & \dots & 0 \\ 1 & 0 & 0 & \ddots & \vdots \\ 0 & 1 & 0 & \ddots & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}$$

C is a nilpotent matrix and $\forall j \in \mathbb{N},\ A^j = \sum_{p=0}^{j} \binom{j}{p} C^p$.

Therefore, for $j \geqslant n-1$:

$$A^j = (\frac{1}{2} - \epsilon)^j \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ \binom{j}{1} & \ddots & \ddots & & \vdots \\ \binom{j}{2} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \binom{j}{n-1} & \dots & \binom{j}{2} & \binom{j}{1} & 1 \end{pmatrix} \tag{3.3.8}$$

and for $j < n-1$:

$$A^j = (\frac{1}{2} - \epsilon)^j \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ \binom{j}{1} & 1 & \ddots & & & & & \vdots \\ \binom{j}{2} & \binom{j}{1} & \ddots & \ddots & & & & \vdots \\ \vdots & \binom{j}{2} & \ddots & \ddots & \ddots & & & \vdots \\ \binom{j}{j} & & \ddots & & \ddots & \ddots & & \vdots \\ 0 & \ddots & & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & \ddots & & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \binom{j}{j} & \dots & \binom{j}{2} & \binom{j}{1} & 1 \end{pmatrix} \tag{3.3.9}$$

Stationary probability distribution

Let Π be the equilibrium distribution of the system, i.e. from the relation 3.3.6, the distribution such that $A\Pi + B = \Pi$ This distribution is also called the stationary distribution of the system

$$\Pi = \begin{pmatrix} \pi_1 \\ \pi_2 \\ \vdots \\ \pi_n \end{pmatrix}$$

we have:

$$A\Pi + B = \Pi \Leftrightarrow \begin{cases} (\frac{1}{2} + \epsilon)\pi_1 = \frac{1}{2} \\ \forall i \in [\![2; n]\!],\ \pi_i = \dfrac{1-2\epsilon}{1+2\epsilon}\pi_{i-1} + \dfrac{2\epsilon}{1+2\epsilon} \end{cases} \tag{3.3.10}$$

It is an arithmetic-geometric sequence which can be solved for all agents i as

$$\begin{cases} \pi_1 = \dfrac{1}{1+2\epsilon} \\ \pi_i = \dfrac{1}{2} + \left(\dfrac{1-2\epsilon}{1+2\epsilon}\right)^{i-1}(\pi_1 - \dfrac{1}{2}) \end{cases} \tag{3.3.11}$$

with

$$\pi_1 - \frac{1}{2} = \frac{1}{2}\left(\frac{1-2\epsilon}{1+2\epsilon}\right)$$

Further, we can write equation (3.3.11) as

$$\begin{aligned} \pi_i &= \frac{1}{2}\left[1 + \left(\frac{1-2\epsilon}{1+2\epsilon}\right)^i\right] \\ &= \frac{1}{2} + \frac{1}{2}\exp\left[-i\ln\left(\frac{1+2\epsilon}{1-2\epsilon}\right)\right] \\ &= \frac{1}{2} + \frac{1}{2}\exp\left[-\frac{i}{\ell_c}\right] \end{aligned} \tag{3.3.12}$$

where ℓ_c is defined as

$$\ell_c = \frac{1}{\ln\left(\frac{1+2\epsilon}{1-2\epsilon}\right)} \tag{3.3.13}$$

and referred to as the *control length* as it gives a value for i above which the exponential falls quickly to zero. It is a characteristic distance from the controlled agent where its influence starts to fade.

We see that, when ϵ approaches $1/2$, the length of control ℓ_c converges to 0, which corresponds to a total loss of the controllability of the system. Figure 3.12 shows that ℓ_c decreases very quickly to 0 when ϵ increases to 1/2.

3.3.3 Average vote of the system

In stationary regime, we can calculate the average density of agents with vote 1.

$$S = \frac{1}{n}\sum_{i=1}^{n}\pi_i$$

when n is the number of "free" agents.

According to (3.3.12), we have

$$S = \frac{1}{n}\left[\frac{n}{2} + \frac{1}{2}\sum_{i=1}^{n}\left(\frac{1-2\epsilon}{1+2\epsilon}\right)^i\right] \tag{3.3.14}$$

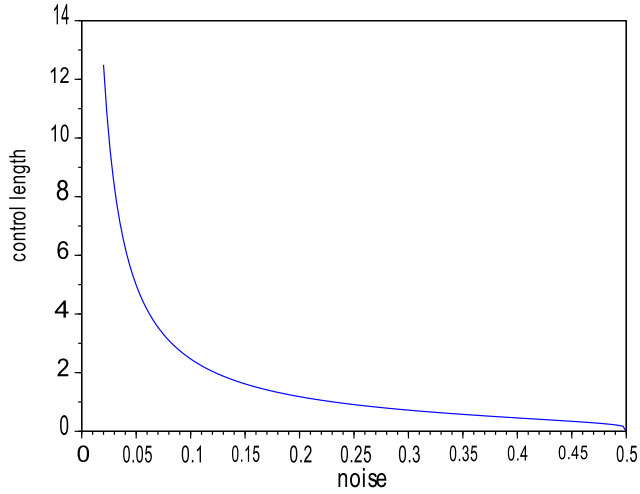

Figure 3.12 – Control length ℓ_c as a function of the noise ϵ, according to equation (3.3.13).

When $\epsilon \neq 0$, we have

$$\sum_{i=1}^{n} \left(\frac{1-2\epsilon}{1+2\epsilon}\right)^i = \left(\frac{1-2\epsilon}{1+2\epsilon}\right) \sum_{i=0}^{n-1} \left(\frac{1-2\epsilon}{1+2\epsilon}\right)^i \tag{3.3.15}$$

$$= \left(\frac{1-2\epsilon}{1+2\epsilon}\right) \frac{1-\left(\frac{1-2\epsilon}{1+2\epsilon}\right)^n}{1-\left(\frac{1-2\epsilon}{1+2\epsilon}\right)} \tag{3.3.16}$$

and we obtain

$$\begin{aligned} S &= \frac{1}{2} + \frac{1}{2n}\left[\left(\frac{1-2\epsilon}{1+2\epsilon}\right) \frac{1-\left(\frac{1-2\epsilon}{1+2\epsilon}\right)^n}{1-\left(\frac{1-2\epsilon}{1+2\epsilon}\right)}\right] \\ &= \frac{1}{2} + \frac{1}{2n}\left(\frac{1-2\epsilon}{4\epsilon}\right)\left[1-\left(\frac{1-2\epsilon}{1+2\epsilon}\right)^n\right] \end{aligned} \tag{3.3.17}$$

In Figure 3.11, we see that the simulations are in agreement with this theoretical result. Indeed, the density of agents who vote 1 oscillates around the mean value S represented by the dashed lines in this figure.

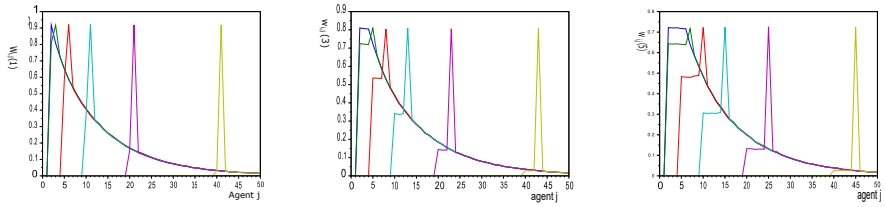

Figure 3.13 – Delayed mutual information, $w_{i,j}(\tau)$, as a function of agent j, for different values of i. The different colored curves correspond to $i = 1,\ 2,\ 5,\ 10,\ 20$ et 40, from dark blue to yellow, respectively. The vote of agent $i = 0$ is forced to 1 and the noise is $\epsilon = 0.01$. The delay is $\tau = 1$ (left panel), $\tau = 3$ (middle panel) and $\tau = 5$ (right panel).

Delayed mutual information

In this section we compute the influence of an agent based on the τ-delayed mutual information, $w_{i,j}(\tau)$, between agents i and j, as defined in eq. (3.1.2). These values are obtained by a sampling of the simulation of the 1D voter model, with $n = 50$ agents. Measurements are performed when the system has reached a stationary state, that is after t iterations such that all the probabilities $A^t P(0)$, corresponding to the free dynamics in equation 3.3.7, are smaller than a certain threshold. In our case, we take the threshold at 10^{-4}.

In Fig. 3.13, we notice that the mutual information $w_{i,j}(\tau)$ is zero if $j < i$, has a plateau for $j < i + \tau$, shows a peak for $j = i + \tau$, and decreases for $j > i + \tau$. This observation reflects the fact that agent i can only influence agents on its right as the voting decision of an agent is based on the state of its left neighbor. The plateau shows the influence of the past $j - i$ iterations. The influence of i over j is maximum for $j = i + \tau$ as it takes τ iterations for the vote of i to travel from i to j. For $j > i + \tau$ the influence is not null, as we are in a steady-state regime.
The results that we have are consistent with those that Brede et al. [16] obtained recently when they studied the dependence between the vote of the first agent and that of the others. They computed analytically the dependance of the average stationary vote of a node on a distance i to the node controlled, and observed that this dependence decreases exponentially with i, as it is the case for the mutual information.
In Fig. 3.14 shows the behavior of the delayed mutual information $w_{i,j}(j - i)$ as a function of j. It suggests the following relationship

$$\forall j > i, w_{i,j}(j - i) = \alpha_i \exp\left[-\lambda_i (j - i)\right] \tag{3.3.18}$$

where α_i and λ_i depend on the noise level, ϵ.

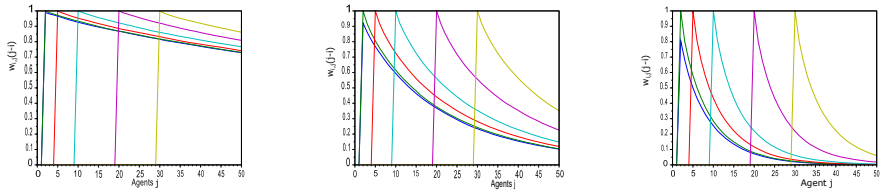

Figure 3.14 – Delayed mutual information $w_{i,j}(j-i)$ as a function of j, for agents $i = 1,\ 2\ 5,\ 10,\ 20$ and 30 (curves from left to right, respectively). The vote of agent $i = 0$ is forced to 1 and the noise is $\epsilon = 0.001$ (left panel), $\epsilon = 0.01$ (middle panel) and $\epsilon = 0.05$ (right panel).

The correlation coefficients between $\ln(w_{i,j}(j-i))$ and j, for different values of the noise, are found to be between -1 and -0.99, thus confirming the relation proposed in equation (3.3.18). The value of α_i and λ_i can be determined with a least squares method.

Consequently, the value of the delayed mutual information $w_{i,j}(j-i)$ decreases quickly as j departs from i. This reflects the difficulty to control agent j from agent i. This interpretation is confirmed by Fig. 3.15 which shows the relation between the values of $\lambda \equiv \lambda_1$ and the control length ℓ_c defined in equation (3.3.13). Each point in this figure corresponds to a different value of the noise. The relation can be fitted by

$$\lambda(\epsilon) = a\frac{1}{\ell_c(\epsilon)} + b \tag{3.3.19}$$

with $a = 0.973$ and $b = -0.003$, independent of the value of ϵ. The coefficient of correlation is 0.997, in agreement with the proposed linear link between λ and $1/\ell_c$.

Noise and information capacity

By evaluating the mutual information, we found that the cost of control increases greatly when the noise increases. This result can be related to the notion of *capacity*, as defined in the standard theory of information. In the linear voter model, agent $i+1$ can be considered as a channel of communication where the input message is the vote of agent i at time t and the output message is the vote of agent $i+2$ at time $t+2$. n this sub-section, we consider that an agent's vote depends only on that of the agent preceding it in the line.

The information channel capacity C_2 is defined as (see [25])

$$C_2 = \sup_{\mathbb{P}_{X_i}} I(X_i(t), X_{i+2}(t+2)) \tag{3.3.20}$$

The capacity value is the maximum amount of information that can pass through the channel.

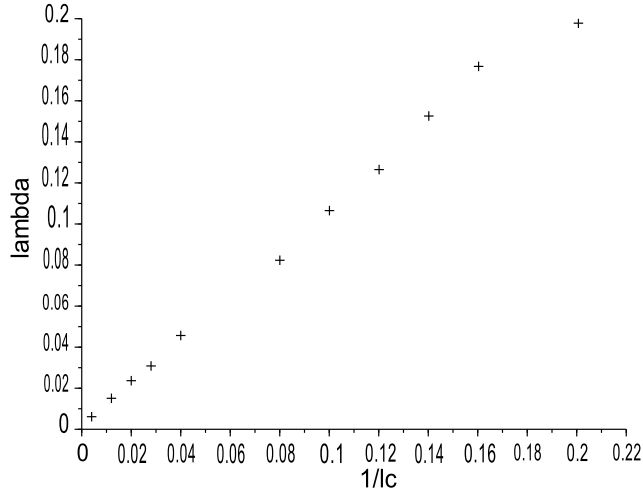

Figure 3.15 – λ as a function of 1/ ℓ_c

Let $P_i(t)$ be the probability distribution of the state of agent i at time t, we have:

$$P_i(t) = \begin{pmatrix} \mathbb{P}(X_i(t) = 1) \\ \mathbb{P}(X_i(t) = 0) \end{pmatrix}$$

we have

$$\begin{cases} \mathbb{P}(X_{i+1}(t+1) = 1) = (1-\epsilon)\mathbb{P}(X_i(t) = 1) + \epsilon\mathbb{P}(X_i(t) = 0) \\ \mathbb{P}(X_{i+1}(t+1) = 0) = \epsilon\mathbb{P}(X_i(t) = 1) + (1-\epsilon)\mathbb{P}(X_i(t) = 0) \end{cases}$$

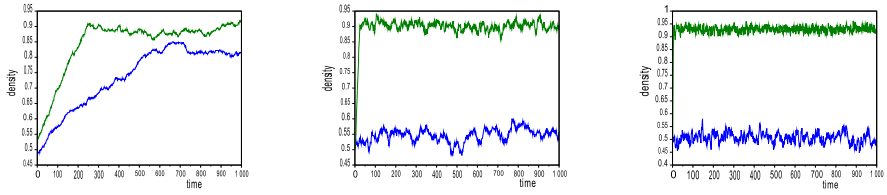

Figure 3.16 – Evolution of the density of vote 1 during the time when only the vote of the agent 1 is forced to 1 (blues curves) and when every vote of the agents $j \propto \lfloor \frac{1}{\lambda} \rfloor$ is forced to 1 (greens curves). left $\epsilon = 0.001$. middle :$\epsilon = 0.01$, right: $\epsilon = 0.05$.

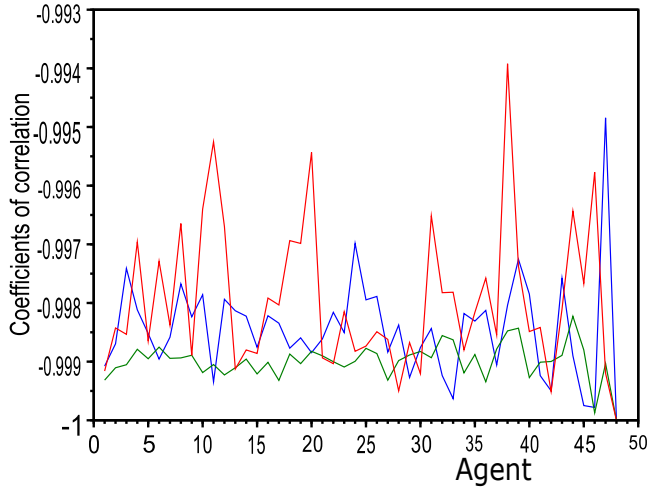

Figure 3.17 – The coefficient of correlation between $w_{i,j}(j-i)$ and i with different noise $\epsilon = 0.001$, 0.01 and 0.05.

which we write as

$$P_{i+1}(t+1) = AP_i(t)$$

with $A = \begin{pmatrix} 1-\epsilon & \epsilon \\ \epsilon & 1-\epsilon \end{pmatrix}$

Therefore,

$$P_{i+2}(t+2) = A^2 P_i(t)$$

where

$$A^2 = \begin{pmatrix} (1-\epsilon)^2+\epsilon^2 & 2\epsilon(1-\epsilon) \\ 2\epsilon(1-\epsilon) & (1-\epsilon)^2+\epsilon^2 \end{pmatrix} \quad (3.3.21)$$

$$= \begin{pmatrix} 1-2\epsilon(1-\epsilon) & 2\epsilon(1-\epsilon) \\ 2\epsilon(1-\epsilon) & 1-2\epsilon(1-\epsilon) \end{pmatrix} \quad (3.3.22)$$

$$(3.3.23)$$

This corresponds to a binary symmetric channel (BSC), with a probability of error

$$p_e = 2\epsilon(1-\epsilon)$$

In the section 2.3, we have described this channel, and we show that the capacity of a binary symmetric channel with a probability of error p_e is

$$C = 1 - H_2(p_e) \quad (3.3.24)$$

with $H_2(p_e) = -p_e \log_2(p_e) - (1-p_e)\log_2(1-p_e)$
Therefore

$$C_2 = 1 - H_2(2\epsilon(1-\epsilon)) \tag{3.3.25}$$

Now, we consider all the agents from $i+1$ to $i+m-1$ as a communication channel between agents i et $i+m$. We denote C_m the capacity of this channel (it depends only on m, the length of the channel). Following the same derivation as before, we obtain

$$P_{i+m}(t+m) = A^m P_i(t)$$

Since A is a symmetric matrix it can be cast in a diagonal form with an orthonormal basis. The eigenvalues are $\lambda_1 = 1$ and $\lambda_2 = 1-2\epsilon$. Thus, A^m can be expressed as

$$A^m = P^T \begin{pmatrix} 1 & 0 \\ 0 & (1-2\epsilon)^m \end{pmatrix} P$$

with $P = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ Thus

$$A^m = \frac{1}{2} \begin{pmatrix} 1+(1-2\epsilon)^m & 1-(1-2\epsilon)^m \\ 1-(1-2\epsilon)^m & 1+(1-2\epsilon)^m \end{pmatrix}$$

and we obtain a symmetric binary channel of length m with a probability of error

$$\epsilon_m = \frac{1}{2}(1-(1-2\epsilon)^m)$$

Therefore, the capacity of this channel is

$$C_m = 1 + \epsilon_m \log_2(\epsilon_m) + (1-\epsilon_m)\log_2(1-\epsilon_m)$$

We know that the capacity is an upper bound of the mutual information. In Figure 3.18, the capacity C_m is shown as a function of its length m, for different values of the noise. The fact that the capacity C_m decreases with m and with the noise, gives another confirmation of the increasing difficulty to control agent $i+m$ by forcing the vote of agent i.
This notion of capacity is another mean of evaluating the effort required to control the system, giving us information on the maximum efficiency to force an agent's in order to influence its closest neighbors vote. It is therefore a way of measuring the cost of control, i.e. evaluating the number of agents whose vote must be forced. The figure 3.18 shows that this cost increases rapidly as a function of the noise, as we can see that the maximum influence of an agent's vote, for noise $\epsilon = 0.05$, decreases exponentially.

Observability analysis in the case of a 1D voter model

The linear voter model analysis given above may be interpreted in terms of reachability or observability Gramians that we introduced in section 2.2.
Such a quantitative information is requested when trying to understand the effects of the distance or noise on the reachability and observability of network systems. Numerous observability or controllability metrics have defined [82], including the reachability and observability Gramians which are measures for the control or observation energies and which are analyzed hereafter. The computation of the reachability Gramians in the case of *target control* problems - where inputs are designed to manipulate a group of target nodes rather than the whole network - is the subject of some recent papers [108, 107]. Qualitative properties and bounds for their eigenvalues have been established [94]. These results apply for general interconnexion topologies and many diffusive dynamical processes, hence also for most voter models. However, in the case of very simple topologies and dynamics - such as in the case of the 1D voter model investigated in this section - it is possible to compute analytically the observability matrix and Gramian and deduce analytical results on the decay of the output signal energy, with the control length and with the noise. It is proved hereafter that one recovers then exactly the behavior exhibited by the delayed mutual information.
Let us consider as previously a 1D topology with $n+1$ voting agents. In section 3.3.2, we mostly considered the case where agent was forced to vote 1. Here we consider a more general case. For $l, m \in \{0, \ldots, n\}$, forcing the vote of agent l is considered as the control action, while observing the vote of agent m is considered as the output measurement. Since we are interested in the deviation from 1/2 of the probability to vote 1 (thus measuring the influence of a forcing action), we define these deviations as state space variables

$$\tilde{p}_i(t) = p_i(t) - \frac{1}{2} \in \left[-\frac{1}{2}; \frac{1}{2}\right] \tag{3.3.26}$$

for all $t \geq 0$ and $i \in \{0, \ldots, n\}$. We will consider in the sequel, with no loss of generality, a forcing of agent 0 vote and an observation of agent n vote, since the influence in the considered linear voter is unidirectional (from left to right). Therefore, the input variable, $\tilde{u}$, and output variable, $\tilde{y}$, will be defined as

$$\tilde{u}(t) = p_0(t) - \frac{1}{2} \; ; \; \tilde{y}(t) = p_n(t) - \frac{1}{2} \tag{3.3.27}$$

Using these state space, input and output variables, the dynamical voter model (3.3.6) transforms into the state space system

$$\begin{aligned} \tilde{\mathbf{p}}(t+1) &= A\tilde{\mathbf{p}}(t) + \mathbf{b}\tilde{u}(t) \\ \tilde{y}(t) &= \mathbf{c}^T \tilde{\mathbf{p}}(t) \end{aligned} \tag{3.3.28}$$

with the state vector $\tilde{\mathbf{p}}(t) = [\tilde{p}_1(t), \ldots, \tilde{p}_n(t)]^T \in \mathbb{R}^n$ and the internal dynamics matrix (generator) A defined as

$$A = \left(\frac{1}{2} - \epsilon\right) \begin{pmatrix} 1 & 0 & \ldots & \ldots & 0 \\ 1 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & 1 & 1 \end{pmatrix} \tag{3.3.29}$$

The control column matrix $\mathbf{b}$ and observation row matrix $\mathbf{c}^T$ are defined respectively as

$$\mathbf{b} = \left(\frac{1}{2} - \epsilon\right) \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ and } \mathbf{c}^T = \begin{pmatrix} 0 & \ldots & 0 & 1 \end{pmatrix} \tag{3.3.30}$$

For any time $t \geq 0$, any initial probability distribution $\tilde{\mathbf{p}}(0) := \tilde{\mathbf{p}}_0 \in \left[-\frac{1}{2}; \frac{1}{2}\right]^n$ and any control (forcing) signal values $\tilde{u}(t) \in \left[-\frac{1}{2}; \frac{1}{2}\right]$, the solution $\phi\left(\tilde{u}; \tilde{\mathbf{p}}_0; t\right)$ of the state space equations (3.3.28) may be written

$$\phi\left(\tilde{u}; \tilde{\mathbf{p}}_0; t\right) = A^t \tilde{\mathbf{p}}_0 + \sum_{j=0}^{t-1} A^{(t-1)-j} \mathbf{b} \tilde{u}(j) \tag{3.3.31}$$

Note that the matrix A has a unique eigenvalue $\lambda(A) = (\frac{1}{2} - \epsilon)$, with multiplicity n and such that $|\lambda(A)| < 1$ (since the noise ϵ satisfies $0 \leq \epsilon < \frac{1}{2}$). Therefore the trajectory (3.3.31) is bounded when $t \to \infty$ and the dynamical system (3.3.28) is said *stable*.

A nonzero state $\tilde{\mathbf{p}} \in \left[-\frac{1}{2}; \frac{1}{2}\right]^n$ is said *unobservable* if the corresponding output can not be distinguished from the output associated with the zero state, that is if

$$y(t) = \mathbf{c}^T \phi\left(0; \tilde{\mathbf{p}}; t\right) = 0 \tag{3.3.32}$$

for all $t \geq 0$ (in the observability analysis, only the free response dynamics is analyzed and $\tilde{u}$ is set to zero). The whole state space system (3.3.28) is said *observable* if the set of unobservable states reduces to $\{0\}$. With the solution (3.3.31) and Cayley theorem, it is easy to prove that this is the case if and only if the *observability matrix*

$$\mathcal{O}_n = \left[\mathbf{c}\ ;\ A^T\mathbf{c}\ ;\ \ldots\ ;\ (A^{n-1})^T\mathbf{c}\right]^T \tag{3.3.33}$$

is full rank or when the *infinite observability Gramian*

$$W^o = \lim_{n \to \infty} \mathcal{O}_n^T \mathcal{O}_n = \sum_{k=0}^{\infty} (A^k)^T \mathbf{c}\mathbf{c}^T A^k \tag{3.3.34}$$

is strictly positive definite.

The infinite observability Gramian gives additionnal quantitative information about how much the system or a particular state is observable. Indeed, the largest observation energy (i.e. the maximum energy for the output signal) is reached when $t \to \infty$ and equals

$$\|\tilde{y}\|_2^2 = \lim_{t\to\infty} \sum_{k=0}^{t} |\tilde{y}(k)|^2 = \tilde{\mathbf{p}}^T W^o \tilde{\mathbf{p}} \tag{3.3.35}$$

for any given state space trajectory $\phi\,(0; \tilde{\mathbf{p}}; t)$. Therefore, with the appropriate change of state space coordinates, the components of the initial condition (or subspaces) may be re-ordered, from the less to the most observable ones. If some of the infinite horizon observability Gramian eigenvalue are zero, then the corresponding vector spaces are unobservable. If some of these eigenvalues are small, then initial conditions variations in the corresponding subspaces will cause low energy variations in the output signal.

In the 1D voter model example, we will analyze to effect of considering the initial probability distribution

$$\tilde{\mathbf{p}} = [1, 0, \ldots, 0]^T \in \mathbb{R}^{n+1} \tag{3.3.36}$$

on the vote of agent $n+1$, by measuring the corresponding observation energy. We will consider a long range time horizon $k > n$ for which the influence of the initial state of agent 1 has reached agent $n+1$ in the line. The last row of matrix A^k may be written (see 3.3.8 and 3.3.9):

$$A^k_{(n+1,\cdot)} = \begin{cases} (\frac{1}{2} - \epsilon)^k \left[\binom{k}{n} \ \binom{k}{n-1} \ \ldots \ \binom{k}{2} \ k \ 1 \right] & \text{when } k \geq n \\ \\ (\frac{1}{2} - \epsilon)^k \left[0 \ \ldots \ 0 \ \binom{k}{k} \ \ldots \ \binom{k}{2} \ k \ 1 \right] & \text{when } k < n \end{cases} \tag{3.3.37}$$

According to definition (3.3.34), since we are measuring the vote of agent $n+1$, we get for the components of the infinite observability Gramian

$$W^o_{i,j} = \sum_{k=0}^{\infty} A^k_{(n+1,i)} A^k_{(n+1,j)} \tag{3.3.38}$$

for all $i, j \in \{1, \ldots, n\}$. Measuring the influence of the initial vote of agent 1, we start with the initial probability distribution (3.3.36) and get, for the agent $n+1$, the observation energy

$$\|\tilde{y}\|_2^2 = \sum_{k=0}^{\infty} \left(A^k_{(n+1,1)} \right)^2 = \sum_{k=0}^{\infty} (\frac{1}{2} - \epsilon)^{2k} \left(A^k_{(n+1,1)} \right)^2 \tag{3.3.39}$$

With equation (3.3.37), one gets

$$\begin{aligned} \|\tilde{y}\|_2^2 &= \sum_{k=n}^{\infty} (\frac{1}{2}-\epsilon)^{2k} \binom{k}{n}^2 \\ &= \frac{1}{(n!)^2}(\frac{1}{2}-\epsilon)^{2n} \sum_{p=0}^{\infty} (\frac{1}{2}-\epsilon)^{2p} \left(\frac{(p+n)!}{p!}\right)^2 \end{aligned} \tag{3.3.40}$$

Using the lower bound

$$(p+1)^n < \frac{(p+n)!}{p!} \tag{3.3.41}$$

one gets

$$\frac{1}{((n-1)!)^2}(\frac{1}{2}-\epsilon)^{2n} \frac{4(3-2\epsilon)}{(1+2\epsilon)^3} < \|\tilde{y}\|_2^2 \tag{3.3.42}$$

On the other hand, since

$$\begin{aligned} &\sum_{p=0}^{\infty} (\frac{1}{2}-\epsilon)^{2p} \left(\frac{(p+n)!}{p!}\right)^2 \\ &\leq \left(\sum_{p=0}^{\infty} (\frac{1}{2}-\epsilon)^{p} \frac{(p+n)!}{p!}\right)^2 \\ = &\left(\sum_{k\geq n} (\frac{1}{2}-\epsilon)^{k-n} k(k-1)\dots(k-(n-1))\right)^2 \\ = &\left(\frac{n!2^{n+1}}{(1+2\epsilon)^{n+1}}\right)^2 \end{aligned} \tag{3.3.43}$$

we get the following upper bound for the observation energy

$$\|\tilde{y}\|_2^2 < \frac{4(1-2\epsilon)^{2n}}{(1+2\epsilon)^{2n+2}} \tag{3.3.44}$$

It is worthwhile to notice how this upper bound behaves with the number of agents along the line and with the noise ϵ. For instance, the upper bound (3.3.44) decreases with the number of agents and the corresponding observation energy is divided by two when k supplementary agents are added in the line, with

$$k \geq \frac{1}{2} \frac{1}{\log_2\left(\frac{1+2\epsilon}{1-2\epsilon}\right)} = \frac{\ell_c}{2} \tag{3.3.45}$$

When the noise increases, the observation energy upper bound decreases

$$\|\tilde{y}\|_2^2 = \mathcal{O}\left((1-2\epsilon)^{2n}\right) \to 0 \text{ as } \epsilon \to \left(\frac{1}{2}\right)^- \tag{3.3.46}$$

The lower bound (3.3.42) decreases similarly, with the same order, when the noise decreases. However it decreases much faster with the number of agents in the voter

line since this lower bound for the observation energy is divided by $\left(\frac{1-2\epsilon}{2n}\right)^2$ when only one agent is added to the n previous ones.

Remark 3.3.1. Note that we performed the observability analysis on the 1D voter model. We could as well develop the dual reachability analysis for the same example. In this analysis, the initial condition is assumed to be zero and one analyzes the forced solution of the state space model (3.3.28).

A reachability Gramian analysis may be used to compute the forcing of agent 1 with minimal energy requested to reach a state $\tilde{\mathbf{p}}$ where all agents in the line vote 1. However, the calculations are much more involved than those performed for observability analysis.
With the solution (3.3.31) and Cayley theorem, it is easy to prove that system is controllable if and only if the *reachability matrix*

$$\mathcal{C}_n = \left[\mathbf{b}\ ;\ A\mathbf{b}\ ;\ \ldots\ ;\ (A^{n-1})\mathbf{b}\right] \tag{3.3.47}$$

is full rank or when the *infinite reachability Gramian*

$$W^c = \lim_{n\to\infty} \mathcal{C}_n\mathcal{C}_n^T = \sum_{k=0}^{\infty} A^k\mathbf{b}\mathbf{b}^T(A^k)^T \tag{3.3.48}$$

is strictly positive definite.
The infinite reachability Gramian gives additionnal quantitative information about how much we can control the system or how much a particular state is controllable. Indeed, the lowest control energy (i.e. the minimal energy requiered to steer the system from state 0 to x_r) is reached when $t \to \infty$ and equals

$$\mathbf{x_r}^T(W^c)^{-1}\mathbf{x_r} \tag{3.3.49}$$

According to definition (3.3.48), we get for the components of the infinite reachabilibility Gramian

$$W^c_{i,j} = (\frac{1}{2}-\epsilon)^2 \sum_{k=0}^{\infty} A^k_{(1,i)}A^k_{(1,j)} \tag{3.3.50}$$

In the equations (3.3.8) and (3.3.9), we have for the first column of matrix A^k:

$$A^k_{(n+1,\cdot)} = \begin{cases} (\frac{1}{2}-\epsilon)^k \left[1\ k\ \binom{k}{2}\ \ldots\ \binom{k}{n-1}\ \binom{k}{n}\right]^T & \text{when } k \geq n \\ (\frac{1}{2}-\epsilon)^k \left[1\ k\ \binom{k}{2}\ \ldots\ \binom{k}{k}\ 0\ \ldots\ 0\right]^T & \text{when } k < n \end{cases} \tag{3.3.51}$$

To calculate the control energy, we need to determine the inverse Grammian of controllability. For a given value of n, we'll use numerical methods to determine W_c^{-1}, and thus deduce this energy.

3.4 Conclusion

In this chapter, we have defined the delayed mutual information and the multi-information, and presented means of calculating them by observing the behavior of a system. The results obtained with the voter model show that these concepts are suitable tools for tackling observability or reachability problems. We showed that time delayed mutual- and multi-informations are promising tools to better grasp the behavior of a dynamical system defined on a complex network. In particular they can be used to determine the most influential or observable agents. This knowledge can be obtained without perturbing the system, by just probing its behavior.

We claim that influential nodes are those that are the most interesting to control or monitor to (i) force a system to reach a given target, or (ii) to have a proxy giving an information on the state of the entire system.

We illustrated our approach with a simple stochastic dynamical model on a graph, the so-called voter model, where agents iteratively adapt their opinion to that of the majority of their neighbors, with however a given noise level. We first discussed the case of a general scale-free topology, where only numerical results can be obtained. Then we consider a 1D topology for which analytical results can be obtained. There, we rigorously showed that the influence of an agent on the entire system can be equivalently measured by actually forcing its behavior, or, in a non-intrusive way, by measuring the time delayed multi-information of this agent with respect to the rest of the system. In particular, we proposed the concept of control length, which indicates a characteristic distance above which the influence of a controlled agent fades exponentially.

The link with classical control theory has been proposed and the control length has been related to the observability Gramian, thus indicating that the observation becomes intractable at large distance. The importance of the noise is clearly shown as being a central element in the possibility of observing or controlling a system, as opposed to some claim in the literature that existence of causality paths was sufficient to achieve control [58].

As an additional link of our approach to existing concepts, we showed that controllability can also be considered in the framework of the capacity of a communication channel, as defined in information theory by Shannon. We showed that this capacity drops as agent are separated by a distance above the control length. In the next chapter we will look at the structural study of a system using information theory.

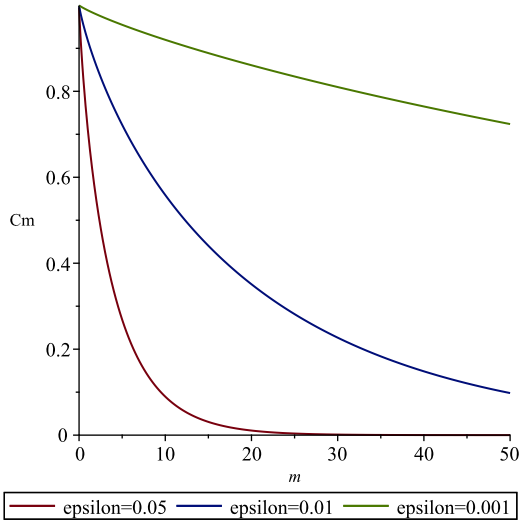

Figure 3.18 – Capacity C_m of the channel between agent i and $i+m$, as a function of m, for different noise levels ϵ.

4 | Topology of a system

4.1 Introduction and review

The knowledge of the interconnection topology of a complex network is important in order to have results on structural observability and reachability, as discussed in [58], [61]. In the section 2.2.4, we presented some results on the subject.
In this chapter we consider the concept of causality as a way to obtain the interaction topology among the variables of complex dynamical systems.
The structural analysis of complex systems dynamics and input-output properties has a long history. Approaches have been developed which make use of interconnection graph or "inference diagram", describing the existing (analytical) relations between a priori given input, state and output variables [56], [100]. Such approaches give structural results on controllability and observability, together with efficient graph algorithms. Some recent results on structural reachability and observability using this approach are presented in [58], [61]. They require a priori knowledge of the system interconnection topology. When trying to isolate the most influential or measurable nodes in some complex system, this information is often missing or incomplete. Besides, these results on reachability and observability only conclude to some existing causal relation between the considered sets of variables, but not how much the dynamics of a node may be measured or controlled from another node.
In many practical situations, the topological structure of a complex network is unknown or uncertain although it plays a pivotal role in its dynamic and control. Therefore, exploring the underlying topology of a complex network has been explored in many papers and some articles propose an information theoretic approach. In neuroscience, for example in the papers [7] and [3], the authors were interested in constructing the graph of interactions between the areas of the brain applied to encephalogram analysis. They identified the topology of this complex system using the transfer entropy [96] and the Granger causality concept [41]. Wilmer et al. [113] used a time-delayed mutual information for detecting nonlinear synchronization in electrophysiological data.

In this chapter, we investigate first in section 4.2 how the delayed mutual information approach may be used to reconstruct the interconnection topology (for instance the adjacency matrix of the interconnection graph). Our goal is to develop an approach which is only based on the sampling of the state dynamics and could be applied online,

for instance with a moving time frame, for the computation of the mutual information. Then, in section 4.3 the problem of a change of the interaction topology during time is investigated. The goal is to detect structural modifications in a dynamical system defined on a graph, by simply observing its state variables. To this end, we compute the delayed mutual-information on sliding time-windows and study whether this quantity can alert us of a change in the structure of the system.
Ideally the approach should be effective when changes occur in the topology, making us able to detect theses changes and reconstruct quickly the new system topology. We will consider - as an illustration example - a probabilistic voter model where the vote dynamics is defined on a scale-free graph representing somehow the influence between agents in a social network.
The quantitative nature of the mutual information and multi-information suggests a new way to define groups or communities in the complex system, based on the information flux between these communities. Partitioning is a traditional problem when developping decentralized approach for the estimation, control or diagnosis purpose (see for instance [80]).

This chapter is organized as follows: section 4.2 indicates how we used our approaches to reconstruct the interconnection topology of a complex system. In section 4.3, we discuss how the delayed mutual-information can be measured in a time sliding-window and how this leads to the identification of dynamical topology changes. Finally, in section 4.4, We will propose in section 4.4 a partitioning algorithm based on this idea and apply it on the examle of the voter model. This partitioning will be the basis for the coarse graining approach developped in chapter 5.

4.2 Identification of the topology

4.2.1 1-Delayed Mutual Information and adjacency matrix

In this subsection, we consider a dynamical complex system whose topology is unknown. The goal is to determine the links between the agents by studying the behavior of the system. The nodes of the graph are known, but the edges are unknown. We observe the state of all agents at each time. This data enable us to identify causal links between agents, which depend on the system dynamics. We wish to construct a graph representing the influence between agents of the system dynamics based on the mutual information between agent's states.

We have seen in the chapter 3 that the delayed mutual information allows us to quantify the influence of an agent on the others. To determine the list of neighbors of an

agent, we need to know its influence on all the other agents with a delay of one time step, and reciprocally the influence of each agent on the one we consider.
Thus, to decide that i and j are neighbors (or not), we will use the values of the $1-$delayed mutual information $w_{i,j}(t,1)$ and $w_{j,i}(t,1)$. We define a threshold T_i depending on the mean value and the standard deviation of the delayed mutual information $\Big(w_{i,j}(t,1)\Big)$, but also depending on the value of the multi-information $w_i(t,\tau)$ of agent i. Indeed, this threshold must be different in the case where one of the agents is a hub because a hub can have a significant influence on agents at a distance of 2 or more. The multi-information allows us to detect the hubs, as we have seen in subsection 3.3.1.
If $w_{i,j}(t,1)$ or $w_{j,i}(t,1)$ are greater than the threshold T_i or T_j, it will be assumed that there is a link between i and j. The overall adjacency matrix of the graph will be reconstructed by examining all pairs (i,j).

4.2.2 Topology of the voter model

In this section, we are interested in the identification of the topology of a voter model with 50 agents. The social network is defined by the graph G represented in figure 4.2. We observe the states of all agents at each instant. The edges of the graph representing the social links are unknown. We wish to determine these edges by observing the behavior of states over time. We will use to use our information metrics to construct this unknown interaction topology. We can see in figure 4.1 the values of the 1-delayed mutual information $w_{i,j}(t,1)$ (see definition 2.3.8) between an agent (for example, in this figure, we have $i = 42$) and all the others in the system. These values were calculated by sampling, when the system has reached its stationary regime. In this case, the highest values of $w_{i,j}(1,1)$ are obtained for the neighbors of agent i. For the example of node $i = 42$ we observed a peak for agents 15, 36, 42, 44 which are indeed neighbors of agent 42 as we can see in figure 4.2.
Thus, we can use the 1-delayed Mutual information to get the edges of a graph. For each agent i, we fixed a threshold T_i for $w_{i,j}(t,1)$ that indicates if j is a neighbor of i or not. This threshold is defined as

$$T_i = \mu_i + a_i\sigma_i$$

where μ_i is the mean of the values of the 1-delayed mutual information between agent i and the others agents, and σ_i is the standard deviation of these values.
There are two possible values for a_i according to the following criteria

$$a_i = \begin{cases} a & \text{if } w_i(t,\tau) > \alpha + \dfrac{1}{2}\beta \\ b & \text{otherwise} \end{cases} \tag{4.2.1}$$

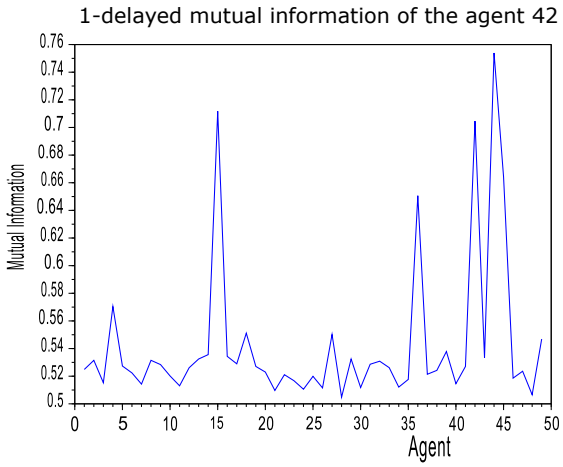

Figure 4.1 – 1-Delayed mutual information between agent $i = 42$ and the rest of the system. Peaks are visible for the neighbors of i.

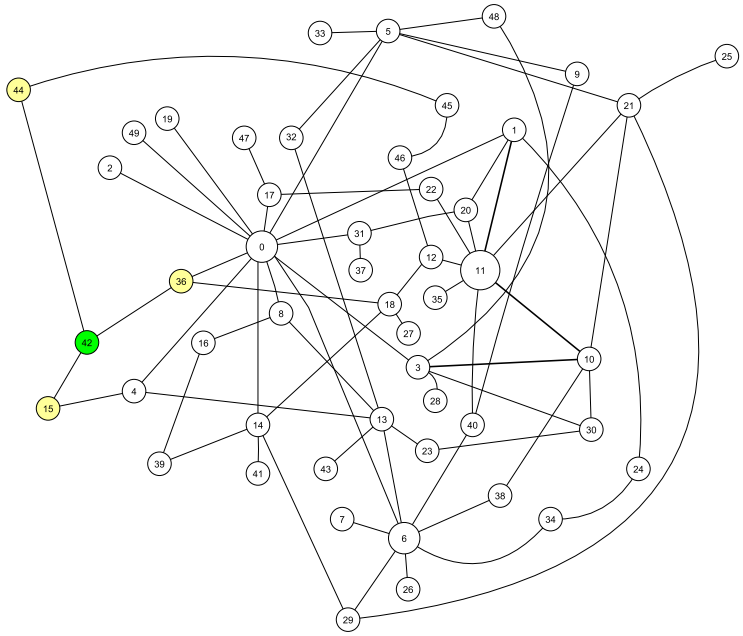

Figure 4.2 – Graph G of the social network from the example in section 4.2.2

where $w_i(t,\tau)$ is agent i's $\tau-$delayed multi-information at time t, as defined in equation (3.1.2). The value of t is chosen to be in the stationary regime and τ must be large enough to allow its influence so spread to all the other agents. The values α and β are respectively the average and the standard deviation of $w_i(t,\tau)$ over i.
The values of (a,b) were determined from an example by optimizing the error rate:

- $\surd$ we took all pairs of (a,b) values between 0 and 1 with a step of 0.1 with $a<b$.
- $\surd$ With these values, we compared the mutual information with the thresholds, and obtained graphs G'_k.
- $\surd$ We compared G_k and G'_k by calculating the error rate $r_k(a,b)$ defined by

$$r_k(a,b)=\frac{\Delta(M_k(a,b),A_k)}{n^2}$$

  where $\Delta(M_k(a,b),A)$ is the Hamming distance namely the number of values that differ between A_k, the adjacency matrix of G_k and $M_k(a,b)$ the adjacency matrix of G'_k. n is the number of agents in the graph.

- $\surd$ $(0.7,0.2)$ is the pair of values of (a,b) for which the error rate over is the lowest.

Next, we randomly generated 20 scale-free graphs $(G_k)_{1\leqslant k\leqslant 20}$, and with the couple of values found, the average error rate obtained is 0.9% and the standard deviation is 0.0026 for these 20 graphs.
The estimation of the adjacency matrix, denoted $M=(m_{i,j})_{1\leqslant i,j\leqslant n}$, is defined by :

$$m_{i,j}=\begin{cases}1 & \text{if } w_{i,j}(t,1)>T_i \text{ or } w_{j,i}(t,1)>T_j\\ 0 & \text{otherwise}\end{cases}$$

In other words, when $w_{i,j}(1)>T_i$ or $w_{j,i}(1)>T_j$, it is assumed that agents i and j are neighbors.

For example, figure 4.3 shows the graph that is reconstructed by this procedure, and compares it to the original graph. In this case, the error rate is $r=1.3\%$. An even better result is obtained when 1-delayed mutual information is computed during the initial transient regime (see figure 4.4). The error rate is then $r=0.24\%$. In the transient regime the results are probably better because the $1-$delayed mutual information really probes the direct influences. During the stationary regime, in contrast long range influences may blur the detection. To provoke such a transient regime, one may disrupt the system by temporarily increasing the noise, while calculating the mutual information. In this example, just 3 edges are missing as we can see in figure 4.4. It seems that these links are not detected because one of the vertices is strongly influenced by another neighbor.

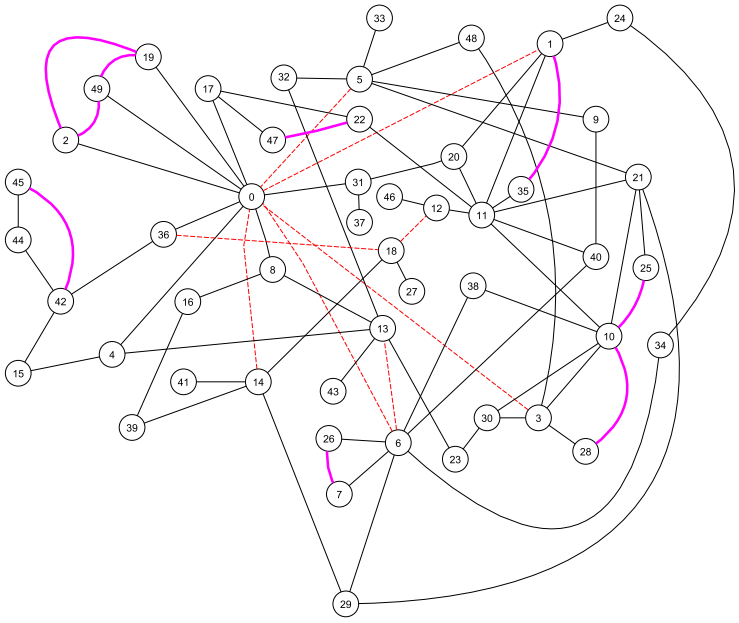

Figure 4.3 – Graph built with the 1-delayed mutual information calculated when the system is in a stationary regime. The dashed red edges are the ones that have not been found, and the solid pink edges are those that were wrongly added.

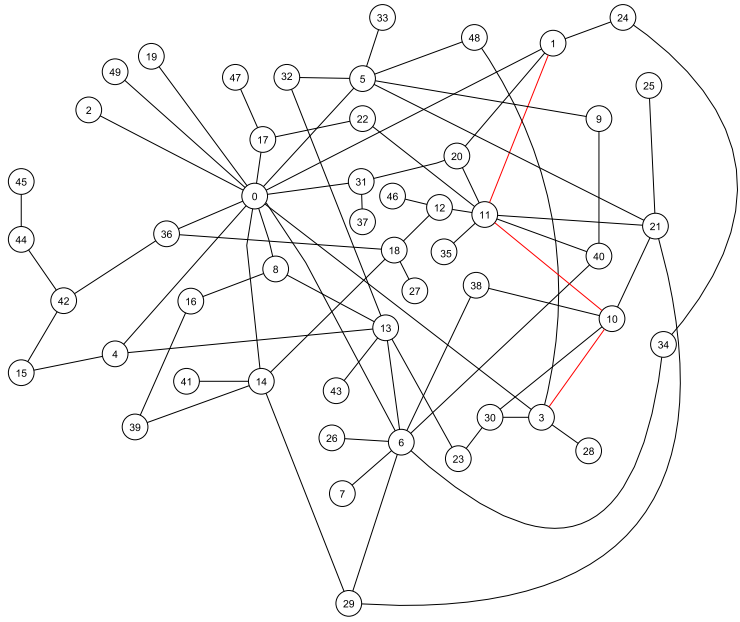

Figure 4.4 – Graph built with the 1-delayed mutual information computed when the system evolves from its initial state. The red edges are the ones that have not been found.

4.3 Change detection

In this section, we are interested in a complex system whose topology changes over time. The goal is to detect these changes by studying the behavior of the states of each vertex. We have seen in the previous chapter that the 1-delayed mutual information allows us to determine the topology of the system. To determine the changes of topology in real time, we evaluate this mutual information on a sliding time window. In a first step, the computational method is presented, and then the results obtained in the case of the voter model are shown.

4.3.1 The delayed mutual information computed on sliding time window

We compute delayed mutual information $w_{ij}(t,\tau)$ at time $t = t_0$ by sampling $s_i(t-\tau)$ and $s_j(t)$ (the states of agents i and j respectively at instants $t-\tau$ and t) on a sliding window $t \in [t_0 - \Delta + 1, t_0]$ with width Δ. During this time interval, we record the pairs $\big(s_i(t-\tau), s_j(t)\big)$ of the states of vertices i and j with a time delay $\tau = 1$ in arrays of integers $N^{k,\ell}$ where k and ℓ describe the set of states. The coefficient $N^{pq}_{i,j}$ of this array is the number of time that $S_i(t-\tau) = k$ and $S_j(t) = l$. According to the definition (2.3.8), the estimation of the $1-$delayed mutual information on the time interval $[t_0 - \Delta + 1, t_0]$ is given by

$$\begin{aligned} w_{i,j}(t_0,\tau) &= \sum_p \sum_q \frac{N^{pq}_{i,j}}{\Delta} \log\Big(\frac{N^{pq}_{i,j}/\Delta}{\big(\sum_r N^{pr}_{i,j}/\Delta\big)\big(\sum_r N^{rq}_{i,j}/\Delta\big)}\Big) \\ &= \sum_p \sum_q \frac{N^{pq}_{i,j}}{\Delta} \log\Big(\frac{\Delta \times N^{pq}_{i,j}}{\big(\sum_r N^{pr}_{i,j}\big)\big(\sum_r N^{rq}_{i,j}\big)}\Big) \end{aligned}$$

In the boolean case, we have only 4 arrays : N^{00}, N^{01}, N^{10} and N^{11}. This leads to the following quantities:

$$\begin{aligned} N^{00}_{i,j}(t_0) &= \sum_{k=0}^{\Delta} \bar{s_i}(t_0 - k - 1) \times \bar{s_j}(t_0 - k) \\ N^{01}_{i,j}(t_0) &= \sum_{k=0}^{\Delta} \bar{s_i}(t_0 - k - 1) \times s_j(t_0 - k) \\ N^{10}_{i,j}(t_0) &= \sum_{k=0}^{\Delta} s_i(t_0 - k - 1) \times \bar{s_j}(t_0 - k) \\ N^{11}_{i,j}(t_0) &= \sum_{k=0}^{\Delta} s_i(t_0 - k - 1) \times s_j(t_0 - k) \end{aligned}$$

with $\bar{s_i}(t) = 1 - s_i(t)$. Therefore we have :

$$w_{i,j}(t_0,\tau) = \frac{N_{i,j}^{00}}{\Delta}\log\Big(\frac{\Delta \times N_{i,j}^{00}}{(N_{i,j}^{00}+N_{i,j}^{01})(N_{i,j}^{00}+N_{i,j}^{10})}\Big) + \frac{N_{i,j}^{01}}{\Delta}\log\Big(\frac{\Delta \times N_{i,j}^{01}}{(N_{i,j}^{01}+N_{i,j}^{00})(N_{i,j}^{01}+N_{i,j}^{11})}\Big)$$
$$+ \frac{N_{i,j}^{10}}{\Delta}\log\Big(\frac{\Delta \times N_{i,j}^{10}}{(N_{i,j}^{10}+N_{i,j}^{11})(N_{i,j}^{10}+N_{i,j}^{00})}\Big) + \frac{N_{i,j}^{11}}{\Delta}\log\Big(\frac{\Delta \times N_{i,j}^{11}}{(N_{i,j}^{11}+N_{i,j}^{10})(N_{i,j}^{11}+N_{i,j}^{01})}\Big)$$

The choice of the value of Δ is a trade-off between two contradictory constraints:

- It must be large enough so that the approximations of the probabilities calculated on the samples of size Δ are not too far from the actual values.
- it must not be too large so that the changes can be detected as soon as possible.

4.3.2 Topology change detection with the voter model dynamics

Detection changes between 2 agents

We consider a graph with $n = 500$ agents, whose topology is modified over time in a prescribed way. The 1-delayed mutual information is computed using a sliding-window with $\Delta = 300$. Figure 4.5 shows the time evolution of $w_{i,j}(t)$ between two selected vertices of low degrees on one side and between two selected hubs (vertices of high degrees) on the other side. Between these pairs of vertices an edge was alternatively added and removed every 2000 time steps. In the case of low degree vertices, we see that the value of $w_{i,j}$ informs us of this change of topology. On the other hand, when the changes occur between the two hubs, $w_{i,j}(t,1)$ does not detect them. However, in this case, it was found that the corresponding edge does not have a great influence on the system dynamics as we can see in figure 4.5 (right).
If the link is modified between a vertex i of low degree and a hub j, $w_{i,j}$ hardly detects this change, as seen in Fig. 4.6 (right). This is expected as a terminal node does not influence a hub. But this change can be detected by measuring the opposite delayed mutual-information, $w_{j,i}$, (see Fig 4.6, left panel), reflecting the fact that the hub does influence a neighboring vertex.
As we saw just before, the 1-delayed mutual information on a sliding window seems to be a good metric to detect when the link between 2 nodes appears or disappears. To confirm that, we made simulations where the topology of the graph changes randomly over time. At each moment, we have a probability p that there is a change. Then, if there is a change, we randomly choose a pair of nodes: either we add an edge, or we remove it. For example, with a scale free graph of order $n = 500$ and with a probability of change $p = 0,01$ we had 54 changes during a simulation over a duration of 6000 iterations. In this simulation all changes have been detected because we have no changes between 2 hubs. For each pair of nodes i and j, we computed the $1-$delayed mutual information on a sliding window of size 100. When these 2

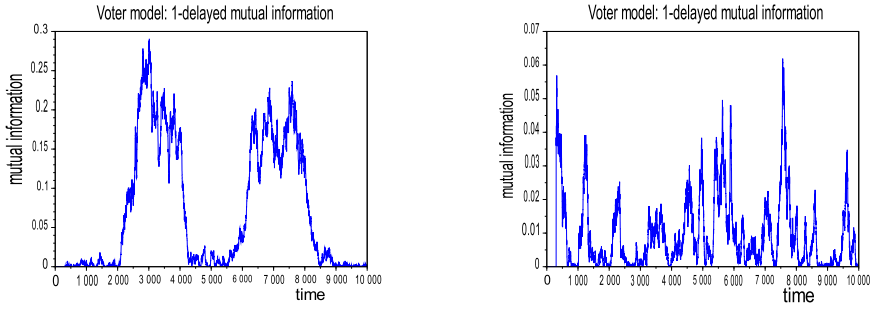

Figure 4.5 – Plots of the 1-delayed mutual information between two vertices of low degree (left) and between two hubs (right) during a change of topology between these vertices at times $2000,\ 4000,\ 6000,\ 8000$. The parameters of the simulation are $\epsilon = 0.01$, $n = 500$, $\Delta = 300$.

values exceed and remain above a threshold, we assume that a new edge has appeared. And when these values become lower than the threshold, we assume that an edge has disappeared. With this method, the average time to detect changes is 160 iterations with a standard deviation of 110.

4.3.3 Conclusions

In this section, we have described a way to detect topology changes in a dynamical system on a graph, such as the voter model. Detecting structural changes is important as it can provide an early warning of tipping points. We expect that our approach can be applied to many other complex systems. The 1-delayed mutual information computed on a sliding-window was used to identify the possible changes of topology. We saw that this quantity allows us to detect whether an edge is added or removed between two vertices of low degree, or between a hub and a vertex of low degree. Between two hubs this method is not effective, but the actual presence or lack of a link between them does not affect much the global behavior of the system. As we'll see in the next section, each hub can be seen as a seed that characterizes its community, so we can assume that its state depends mainly on the states of the nodes in its community, so its vote is unlikely to be influenced by another hub, which would be its neighbor and which would also vote like the members of its community.

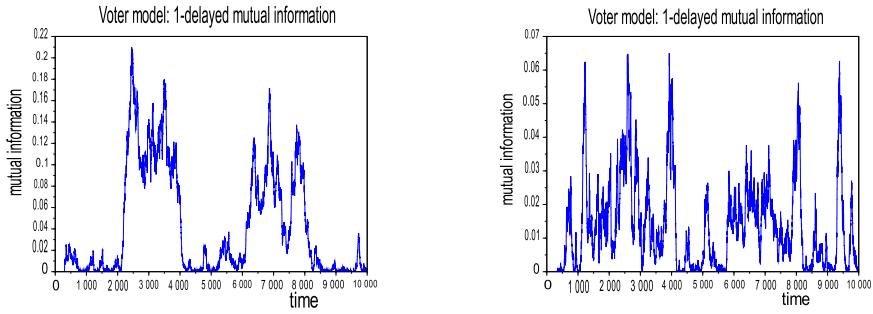

Figure 4.6 – Left: plot of the 1-delayed mutual information between a hub and a low degree vertex. Right: plot of the 1-delayed mutual information between the low degree vertex and the hub. The parameters of the simulation are $\epsilon = 0.01$, $n = 500$, $\Delta = 300$.

4.4 Detection of communities with information theory

4.4.1 Introduction

The existence and structure of communities in a graph is an important concept in the analysis of social networks. Several definitions of communities exist. As we have seen in subsection 2.1.5, the definition of community most often used is: communities as sub-graphs with dense internal connections and sparse connections between these sub-graphs. Our approach is different: we define communities based on system dynamics, which we call communities of influence.
Papadoulos [81], Liu et al.[57], Zhang [118], Fortunato [33] and Newman [74] present in their paper different algorithms for finding communities. They use modularity (see definition 2.1.16) to define communities.
We want to determine a partition of a complex network into communities to determine a reduction of the system. The goal of this reduction is to facilitate the system control. For this reason, we define the notion of community in a specific way. Communities should enable us to control the system, therefore they must be built from the dynamics and not from the topology only. In our case, communities will be identified by those nodes which have an important role in the system dynamics, and called hereafter influencers (see [86]). Communities will be built around influencers which have a significant impact on the system behavior. For instance, opinion leaders in social networks, are capable of influencing the public's point of view on certain topics.[109].

Since we are using scale-free graphs as models for social networks, we have adopted a seed-centric approach for communities (see [49]). Communities will be built around

seeds, which are the influencers of the graph.
In this chapter, we will present an original method based on the observation of the behavior of the dynamical complex system to find influencers and their community. We will use delayed multi-information to determine these seeds, and then define a notion of proximity based on delayed mutual information to build communities around seeds. As in the dynamics of the voter model, the state of a node will depend generally on its neighbors state.
We will use the modularity (see the definition 2.1.16), frequently used to evaluate the quality of partitioning and we will compare our partitions with those obtained by some construction methods using only the system topology: the Louvain algorithm, method that maximizes modularity. To evaluate the quality of our partitions, we will also use Markov Stability, a value that quantifies the time spent by a random walker within communities before moving on to another part of the network. Finally, we will apply our algorithm to a dynamics where the state of a node does not depend directly on its neighborhood to underline the difference between these methods.

4.4.2 Community detection

The purpose of this section is to present a simple method to create a partitioning of the agents of a complex system into the type of communities described above, using information theory tools.
In a first step, we determine the most influential agents which we have named influencers.
We have seen in section 3.1.4 that the delayed multi-information can determine these agents. They will be the seeds of our community partitioning. If we want to partition the agents into k communities, we choose for the seeds the k nodes with the largest multi-information values.
Then, to build the community for each of these seeds agents, we use a proximity notion

Definition 4.4.1. Let $G = (V, E)$ be a graph, the proximity $p(i, j)$ between two nodes i and j is defined as

$$p(i,j) = \frac{1}{r} \sum_{\tau=1}^{r} \omega_{i,j}(1, \tau) \tag{4.4.1}$$

where r is an approximation of the radius of the graph G.

Remark 4.4.2. Generally, $p(i, j) \neq p(j, i)$

This notion of proximity in terms of influence between agents i and j is thus calculated by averaging the delayed-mutual information. As this value takes into account the delayed-mutual information for different τ values, even for nodes far from all

seeds it's possible to determine the closest seed in terms of influence.
In the case where we have a scale free graph, we can use for the radius r the asymptotical $\frac{1}{2}\frac{\log(n)}{\log(\log(n))}$ where n is the order of the graph (see Bollobas [12], based on the notions of influencers and proximity, we can now define communities in terms of system dynamics:

Definition 4.4.3. Let $\{s_1, ...s_k\}$ be a set of k influencers in a complex system. For any $i \in \{1, 2, ...k\}$, the community of s_i is the set of agents C_i whose state over time depends more on s_i than on all other influencers, i.e. $x \in C_i$ if and only if the influencer closest to x is s_i in the sense of the proximity (4.4.1).
$(C_i)_{1 \leqslant i \leqslant k}$ is a partition of the set of agents in the system.

To build communities, for each agent j who is not an influencer, we determine its community, which is that of seed agent i^* , such as

$$i^* = \underset{i \in I}{\operatorname{argmax}}\, p(i, j) \tag{4.4.2}$$

where I is the set of seed agents (influencers).

4.4.3 Detection of the communities of influence in the voter model

We are going to test the algorithm described in the previous section in the case of the voter model described in the section 3.2. First we take a social network structure for which the expected decomposition into communities is obvious: a circular graph of 5 complete subgraph (see figure 4.7).
The value of the $3-$delayed multi-information obtained for agents $1, 11, 21, 31, 41$ and 51 is 0.0054 and for the other agents we obtained the value 0.0035. The $3-$delayed multi-information allows to detect the influencers of the system. Then, the proposed algorithm in the previous section gives us the expected partition in which the communities correspond to the cliques.
Now we take a scale free graph of order 50. We obtained the communities that can be seen in figures 4.8 which shows a coherent partitioning into communities. A graph is considered to have a significant community structure when a partition obtains a modularity score greater than 0.3 (see [44]). The computation of the modularity values Q in table 4.4.4 confirms the visual aspect. For our example of graph, we can see that we are close to the best possible modularity values obtained with the Louvain algorithm. For the dynamic of this voter model, we can see that the communities of influence almost correspond to the topological communities obtained with a standard algorithm.

The number of communities can be chosen judiciously according to the values of the multi-information. In this example, in table 4.1, we can see the largest values for the

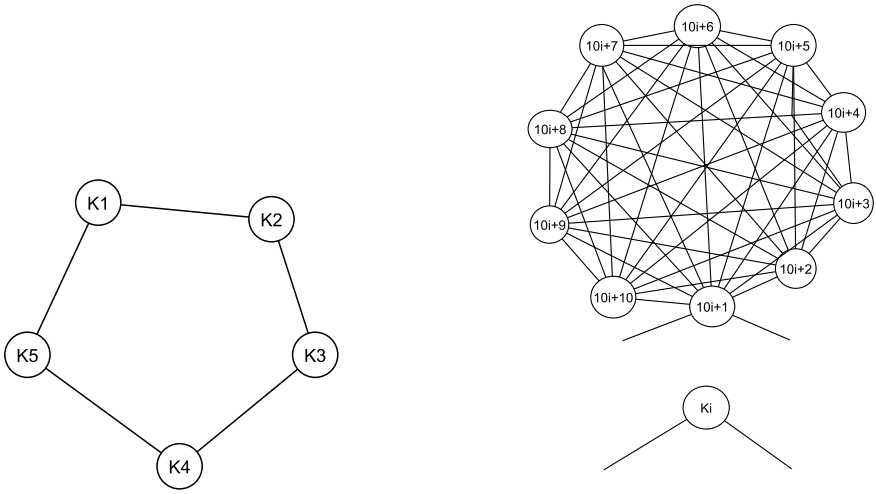

Figure 4.7 – Left: ring graph of 5 complete subgraph of order 10. Right: a complete subgraph K_i of order 10, the vertices of this subgraph are numbered from $10i + 1$ to $10i + 10$

$3-$delayed multi-information. The chosen delay ($\tau = 3$) is approximately equal to the radius of the graph.

In our case, the largest value of the multi-information is approximately three times larger than the second. This value corresponds to the largest community. According to the values obtained, we can see that when we have 3 communities, it corresponds to 39.2% of the sum of the values of the multi-information. This decomposition seems optimal, since when the following influencers are added, the share of influence increases little.

We applied our algorithm for the scale-free graph G_1 shown in figure 2.5. This graph has the particularity of having three agents that have a much greater influence than the other agents, as it may be seen from the largest values of the multi-information in table 4.1.

In figure 4.9, we have the partitions obtained for 3 or 6 communities, we can see that from the 4th most influential node, the cardinals of the communities associated with these influencers are small. In this case we will prefer the partition into 3 communities

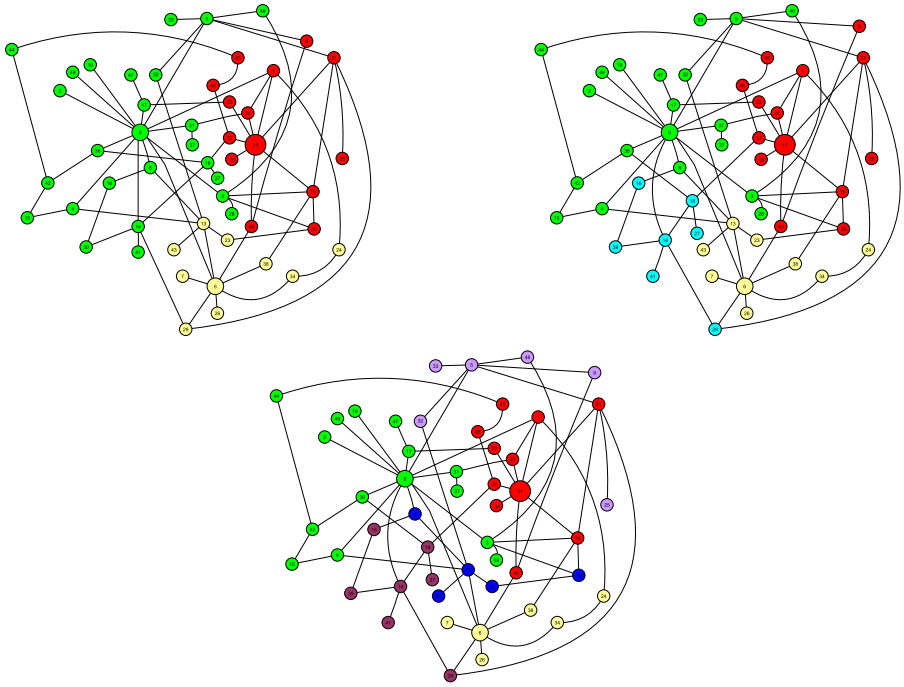

Figure 4.8 – Partition of the graph in 3, 4 and 6 communities obtained with the seeds which are, in descending order of multi-information, agents $0, 11, 6, 14, 5, 13$

because the 4th influencer in the multi-information sense has a much smaller influence than the first three ones.

4.4.4 Comparaison of the communities obtained

To compare the communities obtained with our method, we are interested in two tools: modularity (see definition 2.1.16) and autocorrelation, which we will present bellow in this section.

First, we built the decomposition of this system using the Louvain algorithm [38] in which the communities are obtained by maximizing the modularity. In figure 4.10, we can see the decompositions obtained with this algorithm. This method uses only the topology of the graph, contrarily to the algorithm presented above which builds the communities from the influence of the agents, and which takes also into account the local dynamics. We can notice that there are just a few differences between the partitions obtained with the 2 methods. In figure 4.10, agents belongs to different communities in the 2 methods are surrounded by a circle with the color of their community

i	0	11	6	14	5	13
$w_i(1,3)$	0.047	0.0166	0.0162	0.010	0.0085	0.069
$\frac{w_i(1,3)}{\sum_j w_j(1,3)}$	23%	8.20%	8.03%	5.02%	4.21%	3.43%
$\frac{\sum_{j\|w_j\leqslant w_i} w_j(1,3)}{\sum_{w_j} w_j(1,3)}$	23%	31.7%	39.20%	44.22%	48.43%	51.86%

Table 4.1 – $3-$delayed multi-information of the most influencers and the percentage of influence they have on the system

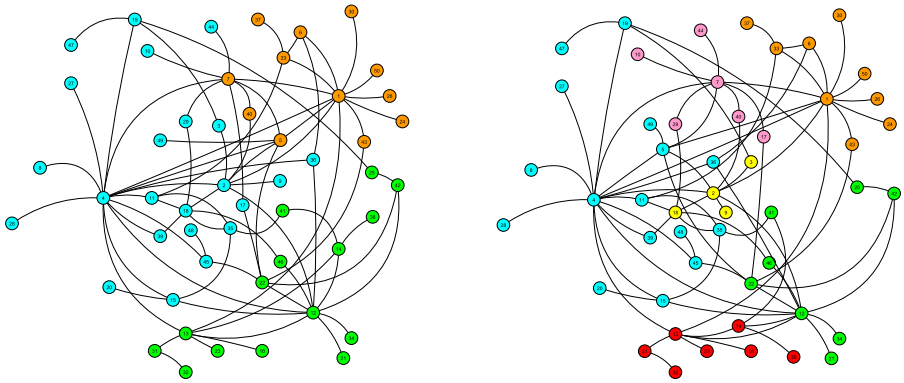

Figure 4.9 – Partition of graph G_1 in 3 and 6 communities.

in the first method.

In table 4.4.4, we can compare the maximum values of the modularity Q_M for the studied graph and the values Q obtained with our method based on the influence of the agents. We can see that the values are close, even though with the first method communities are built also from the dynamic, and in second from the structure only.

Markov Stability [27], [54], allows to evaluate the quality of a community partition in terms of the persistence of Markov dynamics inside the communities. The greater the Markov stability, the less likely it is that a random walker in the graph will escape from a community.

Let be an undirected graph G of order n , with adjacency matrix A. We consider a community partition $\mathcal{C} = \{c_1, c_2, ..., c_k\}$ characterized by a matrix $H \in \mathcal{M}_{n,k}(\mathbb{R})$

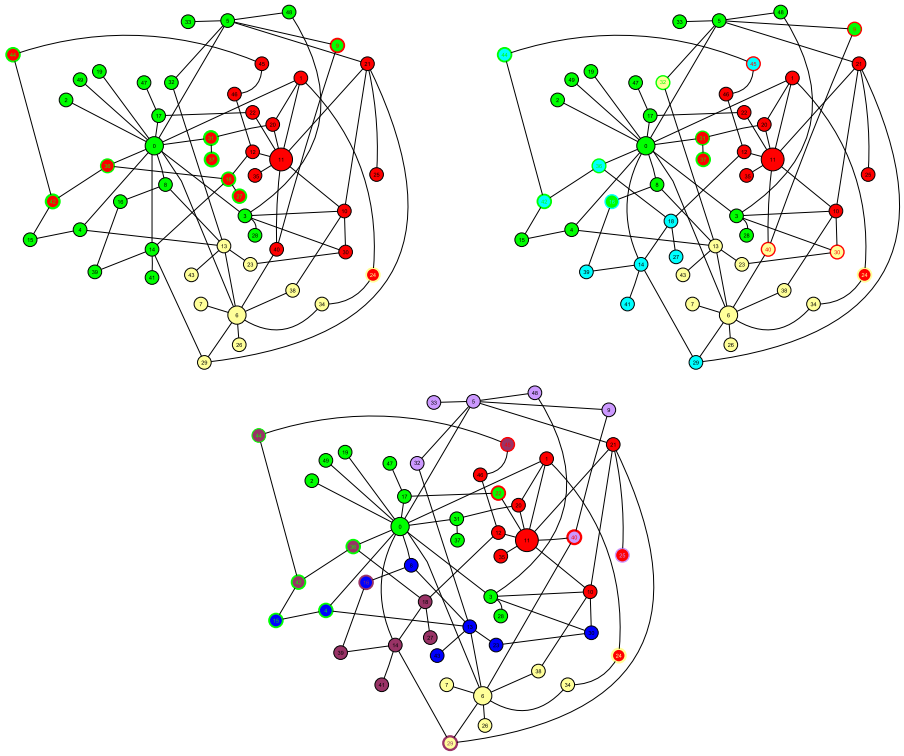

Figure 4.10 – Partition of graph in 3, 4 and 6 communities obtained with the algorithm of Louvain [38], [75] that maximizes the value of the modularity.

defined by

$$H_{i,j} = \begin{cases} 1 & \text{if the nodes i belongs to the community } \ c_j \\ 0 & \text{otherwise} \end{cases}$$

Let $d = (d_1, d_2, ..., d_n)$ be the degree vector, and $D = diag(d)$ the degree diagonal matrix and m is the number of edges. We have $m = \frac{\sum_{i=1}^{n} d_i}{2}$.
We define a discrete-time Markov process by the following dynamics:

$$X_{t+1} = X_t M \tag{4.4.3}$$

with $M = D^{-1}A$ and $X_t = p_1(t), p_2(t), ..., p_n(t))$ with $p_i(t)$ is the probability that the random walker is on node i at time t
For this Markov process, we define the clustered autocovariance matrix $R(t)$ at the time t as:

$$R(t) = H^T(\Pi P(t) - \pi^T \pi)H \tag{4.4.4}$$

Table 4.2 – Modularity for different partitioning.

Number of communities	3	4	6
Q	0.3607	0.4208	0.4382
Q_M	0.4046	0.4594	0.5010

with $\pi = d^T/(2m)$ the stationary probability distribution of the system, $\Pi = diag(\pi)$ and $P(t) = M^t$.
The Markov Stability is the trace of this matrix:

$$MS_t(H) = \text{trace}(R(t)) \tag{4.4.5}$$

The Markov stability value indicates how long a random walk remains confined within the communities defined by a specific network partition. A high stability value means that the random walk remains within the communities for a long time before spreading to other parts of the network. We need therefore to find the partition that maximizes $MS_t(H)$. It is an NP-hard problem ([55], [65]).

In our case we use Markov stability to evaluate the quality of the network structure with the partitions that we have obtained with our algorithm and with Louvain's algorithm. The corresponding Markov stability values MS_1 and MS_2 with $t = 10$ or $t = 100$ are given in table 4.4.4.

Table 4.3 – Markov Stability for different partitioning with $t = 10$.

Number of communities	3	4	6
MS_1	0.0692	0.0804	0.0.0878
MS_2	0.0702	0.084	0.0951

Table 4.4 – Markov Stability for different partitioning with $t = 50$.

Number of communities	3	4	6
MS_1	0.0058	0.0066	0.0091
MS_2	0.00049	0.014	0.0019

In the short term (small t value), a high stability value means that small local communities are well defined. In the long term (large t value), a high stability value indicates that larger community structures remain coherent over longer periods.
The values obtained confirm the results obtained above for modularity.

As already mentionned, the construction of our communities depends also on the dynamics of the system (and not only on the topology). The motivation of this decomposition is to determine a simplification of the network in order to be able to control it more simply. The communities obtained may therefore differ from those obtained by methods based on topology only. For example, let us consider that an agent's vote at time $t+2$ depends on the votes of its first neighbors and second neighbors at time t, according to the same transition function f defined by equation 3.2.4. In this case we have:

$$s_i(t+2) = \begin{cases} 1 & \text{with probability } f(\rho_i(t)) \\ 0 & \text{with probability } 1 - f(\rho_i(t)) \end{cases} \tag{4.4.6}$$

with

$$\rho_i(t) = \frac{1}{|D_i|} \sum_{j \in D_i} s_j(t) \tag{4.4.7}$$

The set D_i is the set of first and second neighbors of agents i. on figure 4.11, we have a partition into 3 communities with this dynamic. We can see that it is very different from those obtained from the topology with Louvain algorithm.

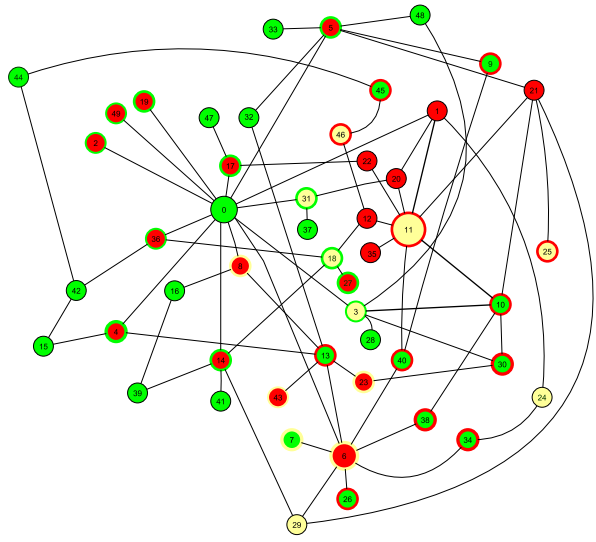

Figure 4.11 – Partition of the graph in 3 communities obtained with our algorithm when an agent's vote depends on the votes of its neighbors and second neighbors.

Consider now another example, where we replace the transition function in equation 3.2.4 with :

$$f(\rho) = a(\rho - 1/2)^2 + \epsilon \tag{4.4.8}$$

where $a = 4(1 - 2\epsilon)$, with the transition function, we have $f(0) = f(1) = 1 - \epsilon$ and $f(1/2) = \epsilon$ In this case, an agent will tend to vote 1 when his neighbors agree, and 0 when the neighbors' vote is equally distributed. As can be seen from the figure 4.12, the communities of influence obtained with this dynamic do not correspond at all to the communities based on the topology. For this partition, the modularity value is $Q = -0.046$. This confirms that the communities of influence we have are topologically unsatisfactory.

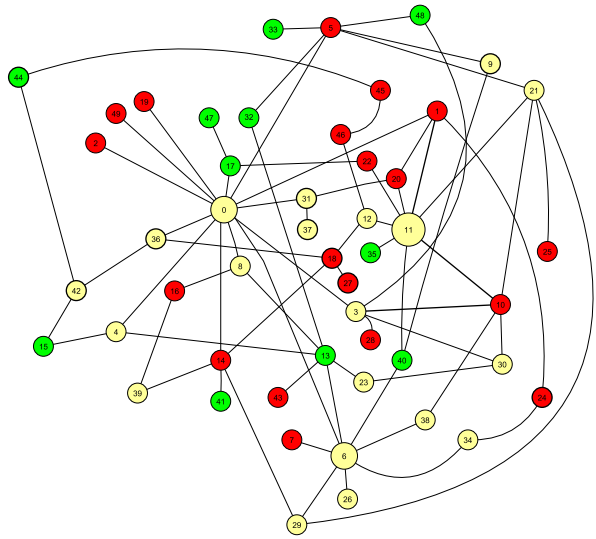

Figure 4.12 – Partition of the graph in 3 communities obtained with our algorithm when with the transition $f(\rho) = 4(1 - 2\epsilon)(\rho - 1/2)^2 + \epsilon$.

4.5 Conclusion

In this chapter, we have seen that mutual information is a tool that can be used to study the structure of a complex system. In the case of the voter model, observation of the system's behavior has enabled us to calculate the delayed mutual information between the agents, i.e. the influence of the agents in relation to each other. Initially, these values, using a heuristic method, were used to find the toplogy of the system. We then looked at a system whose topology changed over time. By calculating the mutual information in real time over a sliding time window, we showed that we were able to determine the modified links that are important with respect to the system dynamics. Finally we have proposed an original and simple algorithm using delayed mutual in-

formation and delayed multi-information to find a partition of the graph into communities. We can build communities by evaluating the influence of agents. These in complex systems can be considered as super agents. Communities may be considered as super-agents and the system dynamics may be reduced, by aggregation, to the dynamics of these super agents. This reduction approach, useful for large-scale complex system control problems, is the subject of chapter 5.

5 | System reduction

5.1 Introduction and review

Many systems of interest to scientists are made up of a large number of interacting components, whose detailed behavior can prove complicated when we wish to observe the system on a larger scale. For instance a fluid is made of many molecules but such a fine grain description is not tractable if one is interested in local macroscopic properties such as pressure, velocity or temperature.
In the realm of complex systems, it is an interesting question to understand when and how a system can be reduced by projecting the fine degree of freedom. We are for instance concerned with the question of the reachability of a complex system, namely our ability to act on it to force its behavior towards a desired objective. Dealing with a large number of degrees of freedom makes this question very challenging, both mathematically and numerically. Therefore, there is a great interest to consider the control of a reduced system as a good approximation of the control of the full system. It is worthwhile to note that the problem has already been investigated in the control community, but mainly for the reduction of large scale unstructured linear systems, i.e. described by a large set of differential (or difference) equations. The proposed reduction methods then rely usually on matrix decomposition algorithms and projections which preserve some system properties such as reachability (see for instance [4] for a textbook on this approach). However these methods do not apply (in general) to nonlinear systems and do not make use of the information on the interaction topology (or *structure*) of the system. In this chapter, we would rather consider reduction methods more specifically adapted to complex systems, relying on local dynamics and local interactions topology (agents neighborhood).
The typical problem we have in mind is a stochastic dynamical system on a graph, where each node is an agent that can have a finite number of possible states. Below we will use our version of the voter model (see chapter 3.2). From a general point of view, our goal is to act on some agent, to impose their state so as to produce a desired global response (e.g. when all agents reach the same state). Intuitively, it seems reasonable to assume that the behavior of the system could be obtained by a reduced number of representative agents that aggregate agents into groups that can be communities.

In what follows, we first explore the above question in a simplified topology of agents,

namely a periodic one-dimensional system, with only nearest neighbors interactions and two possible states. Such a system corresponds to a probabilistic Cellular Automaton (CA). In short, we want to build "supercells" that aggregate cells of the original system, and see whether we are able to express an evolution rule for a global property of these supercells, for instance the state of the majority of the internal cells. First, in the case of cellular automata, we chose to group the cells by 3, which makes it easy to apply the majority rule to the supercells obtained. Secondly, we also propose a reduction of the voter model where the social network is represented by a scale-free graph. In this case, the super cells are the communities identified in the previous chapter.

The process of coarse graining a CA has been addressed by a few authors (see for instance [45, 29, 24]). The main difference of the present approach is that we consider probabilisitc rules for which any projection of the state of a supercell to $\{0, 1\}$ is possible and will give rise to a new probabilistic rule. In the case of deterministic CA as addressed in the literature [114], the challenge is to find whether a projection exists so that the coarse-grained CA is still a deterministic CA. Here, the questions are rather to decide whether the coarse-grained probabilistic CA is a good approximation of the original one (deterministic or probabilistic) and what *good approximation* means in this context.
The chapter is organised as follows. First we develop the formalism for coarse graining a probabilistic CA, and we discuss its applicability in practical cases. Then, we consider the specific case of a CA voter model, its reduction and our capability to design a control strategy on the coarse grained model that would produced the same effect as on the fully resolved CA. We conclude the discussion with open questions and tentative approaches to address them.

5.2 Reduction of Cellular Automata

5.2.1 Problem formulation

The goal of this section is to provide a general framework to define what we mean by the coarse graining of a CA, or its reduction. We start from a general formulation of a 1D CA model. It is well known that the deterministic, elementary CAs can be specified using the numbering scheme proposed by Wolfram [115]. All possible input configurations are listed and the set of corresponding output is interpreted as a number characterizing the rule of CA. This is sketched below for a two-state, radius 1 rule $11011000_2 = 216_{10}$:

$$\underbrace{111}_{1}\ \underbrace{110}_{1}\ \underbrace{101}_{0}\ \underbrace{100}_{1}\ \underbrace{011}_{1}\ \underbrace{010}_{0}\ \underbrace{001}_{0}\ \underbrace{000}_{0} \tag{5.2.1}$$

The output $p = 0$ or $p = 1$ can be interpreted as the probability p that the site $s_i(t + 1) = 1$ provided the values $s_{i-1}s_is_{1+1}$ at the previous time t. Therefore, using the same approach, we can define a stochastic $k = 3$ states and $r = 1$ radius as

$$\underbrace{111}_{p_7} \quad \underbrace{110}_{p_6} \quad \underbrace{101}_{p_5} \quad \underbrace{100}_{p_4} \quad \underbrace{011}_{p_3} \quad \underbrace{010}_{p_2} \quad \underbrace{001}_{p_1} \quad \underbrace{000}_{p_0} \tag{5.2.2}$$

where p_ℓ is the probability that configuration $\ell = s_{i-1}s_is_{i+1}$ is 1 at the next iteration. Therefore any 1D $r = 1$, $k = 3$ CA, whether stochastic or deterministic can be specified by the vector

$$p = (p_7, p_6, p_5, p_4, p_3, p_2, p_1, p_0)$$

with $p_i \in [0, 1]$.

Note that the number of probabilities p_i can be reduced if the rule has symmetries or additional properties. For instance, for the so-called totalistic rules, where the outcome depends only on the sum of the state of the neighborhood, $p_6 = p_5 = p_3$, $p_4 = p_2 = p_1$, and only four components are needed to specify the rule, e.g. p_7, p_6, p_4 and p_0. The number of components of p is also reduced if the rule is symmetric by permuting the left and right neighbors, or by the transformation $0 \leftrightarrow 1$.

The reason why we are introducing this representation is that the coarse graining procedure that we will present transforms a probability vector p into another one, p', by adding noise. The difference between p and p' indicates how the coarse grained rule differs from the original one. Our formalism allows us to consider both deterministic and stochastic rule within the same framework.

5.2.2 Coarse Graining procedure

Our coarse graining procedure is based on several steps that we illustrate on a $r = 1$ CA whose evolution rule is defined through the transition function (or conditional probability)

$$P(s_i(t + 1)|s_{i-1}(t), s_i(t), s_{i+1}(t)) \tag{5.2.3}$$

Generalization to a larger neighborhood or to any kind of interaction graph is left for further investigations. In subsection 5.2.3, we will focus on a scale-free graph.

Figure 5.1 sketches the process. The cells are grouped into super cells of size 3. A given configuration of 3 such supercells produces one configuration of the central super-cell after 3 iterations of the fine-grained original CA (operation T). This super-cell state can then be reduced to $w \in \{0, 1\}$ according to some projection F. The question is then to determine the rule T_{CG} that would lead to the same result by directly projecting the initial super-cells states. As suggested in Fig. 5.1, there might be different sets of super-cells that have the same projection x, y, z, but a different projected output w. This variability can be considered as a probabilisitic component of the reduced CA. Below we detail the steps required for the coarse grainng.

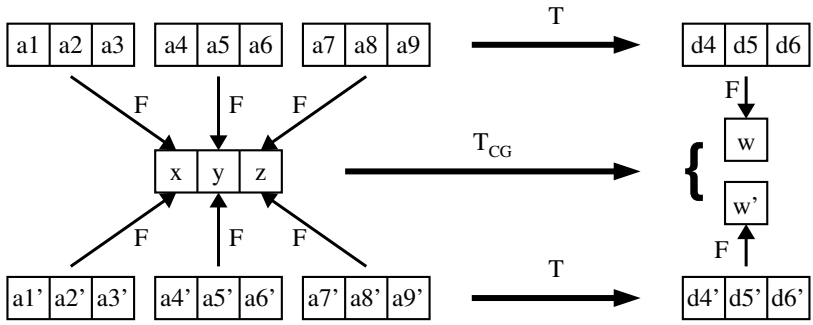

Figure 5.1 – Schematic illustration of the coarse graining procedure for stochastic CA with radius 1.

(1) extension of the CA rule by combining several time steps on the corresponding extended neighborhood. For instance, a $r = 1$, CA, iterated 3 times required the knowledge of the state, $a_1 \dots a_9$, of 9 consecutive cells and determines the state, $d_4\, d_5, d_6$, of 3 cells as illustrated in the following example

$$\begin{array}{lccccccccc} t=0: & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 \\ t=1: & & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 & b_8 & \\ t=2: & & & c_3 & c_4 & c_5 & c_6 & c_7 & & \\ t=3: & & & & d_4 & d_5 & d_6 & & & \end{array} \tag{5.2.4}$$

We have $T(a_1, a_2, ..., a_8, a_9) = (d_4, d_5, d_6)$.
Wolfram's rules, such as rule 216 described above in 5.2.1 , allow us to deterministically obtain the values b_i at time 1 knowing $a_{i-1}a_i$ and a_{i+1} at time 0 for $i \in \{2, 3, ..., 8\}$, the values c_i at time 2 knowing $b_{i-1}b_i$ and b_{i+1} at time 1 for $i \in \{3, 5, ..., 6\}$, and finally the values d_i at time 3 knowing $c_{i-1}c_i$ and c_{i+1} at time 0 for $i \in \{4, 5, 6\}$. The lookup table corresponding to this extension amounts to computing the conditional probability

$$P(d_4, d_5, d_6 | a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) \tag{5.2.5}$$

Such quantities can be obtained from the original rule, namely relation (5.2.3). First

we notice that conditional probability (5.2.5) can be expressed as

$$\begin{aligned} &P(d_4, d_5, d_6|a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) = \\ &\sum_{c_3,\ldots,c_7,b_2\ldots b_8} P(d_4, d_5, d_6|c_3, c_4, c_5, c_6, c_7) \times \\ &P(c_3, c_4, c_5, c_6, c_7|b_2, b_3, b_4, b_5, b_6, b_7, b_8) \times \\ &P(b_2, b_3, b_4, b_5, b_6, b_7, b_8|a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) \end{aligned} \tag{5.2.6}$$

Each of these terms can be obtained from the original CA rule as, for instance for $P(b|a)$,

$$\begin{aligned} &P(b_2, b_3, b_4, b_5, b_6, b_7, b_8|a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) = \\ &P(b_2|a_1, a_2, a_3) \times P(b_3|a_2, a_3, a_4) \times \\ &P(b_4|a_3, a_4, a_4) \times P(b_5|a_4, a_5, a_6) \times \\ &P(b_6|a_5, a_6, a_7) \times P(b_7|a_6, a_7, a_8) \times \\ &P(b_8|a_7, a_8, a_9) \end{aligned} \tag{5.2.7}$$

because, for each cell i, the CA rule is applied independently (new random numbers are drawn for a stochastic rule) and can be computed explicitly with eq. (5.2.3). A similar expression holds for $P(c|b)$. Expression (5.2.6) together with (5.2.7) is analytically heavy but can easily be computed numerically from the original CA rule (5.2.3). In case the original CA is with two states $\{0, 1\}$, one has, using representation (5.2.2), that

$$P(1|xyz) = p_{xyz} \qquad P(0|xyz) = 1 - p_{xyz} \tag{5.2.8}$$

where the notation xyz means the binary number obtained with bits x, y and z. Note the transformation proposed here is exact and contains no approximation. Using the extended rule $P(d|a)$ simply reduces the number of iterations by 3 compared to the original rule. This transformation is denoted T in Fig. 5.1.

(2) Projection of the extended CA rule: this step of the procedure consists in a projection of the extended CA on a reduced one. First one notices that we can group the 9 cells of the extended CA in 3 super-cells. If the original CA has two states, 0 and 1, the super-cells have 3 bits, that is 8 possible states.
We will now define a projection F from these 8 states to $\{0, 1\}$ so that the projected states take the same values 0 and 1 as the original CA (see Fig. 5.1, left, where it is shown that two different state vectors a and a' for 9 contiguous cells are projected through F on the same state (x, y, z) for the three corresponding super-cells).
A simple choice for F, in the case of super-cells with 3-bit states, is the majority rule

$$F(x, y, z) = \begin{cases} 1 & \text{if } x + y + z \geq 2 \\ 0 & \text{otherwise} \end{cases} \tag{5.2.9}$$

This choice will be used in the sequel of the chapter, although the proposed reduction method may be applied using any map F with values in $\{0, 1\}$. The reduced, or coarse grained, CA is then defined by the transition probabilities $P(w|xyz)$ given by

$$P(w|xyz) = P(F(d_{4:6}) = w | F(a_{1:3}) = x, F(a_{4:6}) = y, F(a_{7:9}) = z) \quad (5.2.10)$$

where $\alpha_{p:q} = (\alpha_p, \alpha_{p+1}, \ldots, \alpha_q)$. It is represented as the transition T_{CG} in figure 5.1. These transition probabilities may be written with the joint probability distribution $P(d, a)$ as

$$P(w|xyz) = \frac{P(w, xyz)}{P(xyz)} \quad (5.2.11)$$

$$= \left(\sum_{\substack{F(d_4 d_5 d_6) = w \\ F(a_1 a_2 a_3) = x \\ F(a_4 a_5 a_6) = y \\ F(a_7 a_8 a_9) = z}} P(d, a) \right) \left(\sum_{\substack{F(a'_1 a'_2 a'_3) = x \\ F(a'_4 a'_5 a'_6) = y \\ F(a'_7 a'_8 a'_9) = z}} P(a') \right)^{-1} \quad (5.2.12)$$

where a is short for $a_1, a_2 \ldots a_9$, a' is short for $a'_1, a'_2, \ldots a'_9$ and d is short for d_4, d_5, d_6. Expressing the joint probability $P(d, a)$ as $P(d|a)P(a)$ gives

$$P(w|xyz) = \sum_{\substack{F(d_4 d_5 d_6) = w \\ F(a_1 a_2 a_3) = x \\ F(a_4 a_5 a_6) = y \\ F(a_7 a_8 a_9) = z}} \left(P(d|a) \left[\frac{P(a)}{\sum_{\substack{F(a'_1 a'_2 a'_3) = x \\ F(a'_4 a'_5 a'_6) = y \\ F(a'_7 a'_8 a'_9) = z}} P(a')} \right] \right) \quad (5.2.13)$$

The term $P(d|a)$ is known from the previous step, but the term

$$\frac{P(a)}{\sum_{\substack{F(a'_1 a'_2 a'_3) = x \\ F(a'_4 a'_5 a'_6) = y \\ F(a'_7 a'_8 a'_9) = z}} P(a')} \quad (5.2.14)$$

is unknown as the probability to get the sequence a may depend on the history of the evolution. This probability can be, for instance, estimated by sampling. However, assuming that all configurations are equally likely, (5.2.14) can be obtained easily with combinatorial arguments. This is done in the next section for a deterministic CA example which is known to have a uniform configurations distribution. In section 5.2.3, a probabilistic voter model will be reduced. In this latter example, on the contrary, it is shown both through numerical (empirical) experiments and from adjacency matrix analysis, that the configurations may not be considered as uniformly distributed. In that case, a corresponding weighted reduction has to be proposed.

Illustrative deterministic examples

To illustrate the proposed reduction approach, we first approximated the elementary deterministic chaotic Wolfram rule 30 by a stochastic coarse-grained CA. We assume that the probabilities $P(a)$ are uniformly distributed for all configurations $a = (a_1, a_2, \ldots, a_9)$ We apply the coarse graining approach, we obtain the following probabilities:

$$\begin{aligned} p_0 = 0.6562\,,\ p_1 = 0.5469\,,\ p_2 = 0.6719\,,\ p_3 = 0.7344\,, \\ p_4 = 0.3438\,,\ p_5 = 0.4531\,,\ p_6 = 0.3281\,,\ p_7 = 0.2656 \end{aligned} \tag{5.2.15}$$

Figure 5.2 illustrates the space-time diagram for the exact and approximated blockwise models, starting from the same arbitrary initial configuration. We may notice that, qualitatively, the behaviors of the exact projected model (deterministic) and the coarse-grained approximated model (stochastic) look very similar. Among others, they both exhibit typical patterns issued from the chaotic dynamics. We would like to propose a quantitative comparison of systems with delayed mutual information to evaluate the influence of blocks on each other, the method used will be detailed below. However, in this case, the values of the delayed mutual information between neighboring blocks are so low that it is impossible to use it as a quantitative measure of the reduction error. The mutual information values obtained by sampling are below the absolute stochastic error bound obtained with the central-limit theorem and therefore not meaningful. On the other hand, it is also unclear – in the case of rule 30 – whether or not the choice of the majority projection for the projection F is optimal (or simply meaningful). We may conjecture that the optimal choice for this projection function depends on the specific chosen rule.

Let us now consider another example: Wolfram rule 232.

Rule 232 is a deterministic CA which follows the majority rule. For this rule, we can test the assumption that the probabilities $P(a)$ are uniformly distributed for all configurations $a = (a_1, a_2, \ldots, a_9)$. We will make use, for the projection F, of the majority rule (5.2.9), which is the more natural choice in this example. It is then quite straightforward to prove that the probabilities (5.2.14) are all equal to 1/64. According to expression (5.2.13), to obtain probabilities p_i, we need to compute $P(d \mid a)$. These values are deduced from 5.2.6, 5.2.7 and from the rule 232. We then obtain the following (reduced) transition probabilities for the stochastic approximation:

$$\begin{aligned} p_0 = 0\,,\ p_1 = 0.0312\,,\ p_2 = 0.8594\,,\ p_3 = 0.96688\,, \\ p_4 = 0.0312\,,\ p_5 = 0.1406\,,\ p_6 = 0.9688\,,\ p_7 = 1 \end{aligned} \tag{5.2.16}$$

In order to analyze the error between the exact and coarse-grained dynamics for the blocks, we computed $w_{i,i}$ the 1-delayed mutual information between a block of 3 cells and itself, and then $w_{i,i+1}(1,1)$ the 1-delayed mutual information between the same

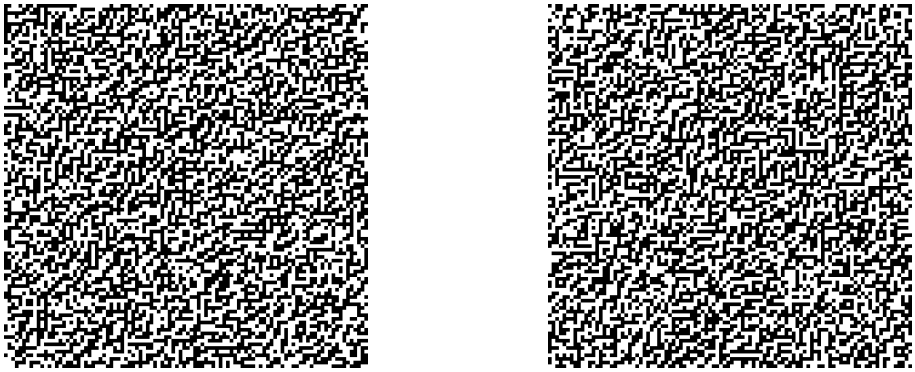

Figure 5.2 – Left: Space time diagram for Wolfram rule 30, computed block-wise for blocks of 3 consecutive agents. The individual cell state values are computed exactly (using rule 30). Then the block state values (black/white) are determined according to the majority projection F. The initial configuration is arbitrary. Right: the same block state values are represented for the proposed coarse-grained elementary stochastic cellular automaton with probabilities given in (5.2.15).

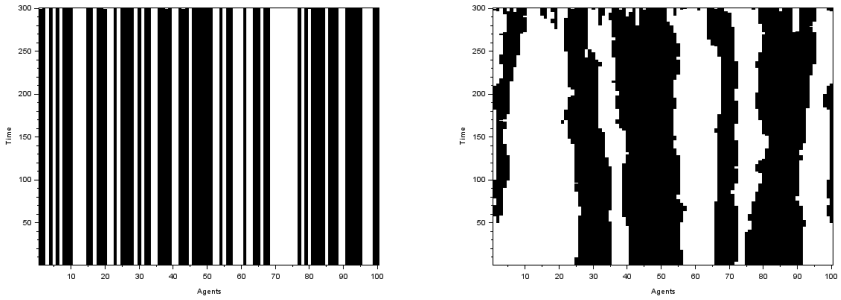

Figure 5.3 – Left: Space time diagram for Wolfram rule 232, computed block-wise for blocks of 3 consecutive agents. The individual cell state values are computed exactly (using rule 232). Then the block state values (black/white) are determined according to the majority projection F. The initial configuration is arbitrary. Right: the same block state values are represented for the proposed coarse-grained elementary stochastic cellular automaton with probabilities given in (5.2.16).

block at the time $t = 1$ and its (right) neighbor. For the exact block-wise model, we have obtained:

$$w_{i,i}(1,1) = 0.71 \qquad w_{i,i+1}(1,1) = 0.004$$

while, for the coarse-grained approximation, we obtained:

$$w_{i,i}(1,1) = 0.67 \qquad w_{i,i+1}(1,1) = 0.004$$

By calculating these values, we can see that the coarse-grained stochastic model obtained with the majority projection F is fairly representative of rule 232, provided we are interested in the dynamic representation of the mutual influence between blocks. Therefore, from the control point of view, the stochastic reduced model seems satisfying because the influence of one block on another, quantified by delayed-mutual information, seems to be preserved. However, when we look at figure 5.3 – which illustrates a scenario where no agent or block are forced (uncontrolled case) – we may observe that the exact coarse grained deterministic and approximated stochastic CA have qualitatively different behaviors. In particular, stochastic variations in the coarse-grained approximated CA may have a significant long range effect on the states of the neighboring cells.

A systematic study of the proposed reduction approach for all elementary Wolfram CAs, with specific projection functions for each of the rules, and suitable metrics to analyze the error dynamics, is still needed and left for future research. This method, which approaches determinist automata with stochastic automata, may prove unsuitable, as in the case of rule 232, where a constant stationary state is obtained. For rule 30, on the other hand, the chaotic character is preserved. This work was initially motivated by control applications where the mutual information is an appropriate measure to evaluate whether or not the approximate system may be used for control design (see section 3.3). Besides, the proposed coarse-grained method was developed to approximate stochastic CAs such as the voter model which is the object of the next section.

5.2.3 The reduced voter model

Description of the model

We will now apply the reduction method to the voter model defined in section 3.2. This agent-based model may be defined on a graph of arbitrary topology. For the sake of convenience, we repeat here the model, binary agent occupies each node of the network. The dynamics is specified by assuming that each agent i looks at every other agent in its neighborhood, and counts the percentage ρ_i of those agents which are in state $+1$ (in case an agent is linked to itself, it obviously belongs to its own neighborhood). A function f, with $0 \leq f(\rho_i) \leq 1$, gives the probability for agent i to

be in state $+1$ at the next iteration. For instance, if f would be chosen as $f(\rho) = \rho$, an agent for which all neighbors are in state $+1$ will turn into state $+1$ with certainty. The update is performed synchronously over all n agents. Formally, this dynamics for the voter model was described in section 3.2.

In this section, we first apply the method seen previously with CAs to the 1D voter model: in this case, we also obtain super cells by grouping 3 agents, and we also use the majority rule for projection. We will determine the transition function of the reduced model, and use mutual information to compare the projected model with reduced model. Finally, we will reduce the voter model defined on a scale-free graph. We will use the community partitioning found in section 2 (just the case with 3 communities). Each community will be merged to form super cells.

Transition probabilities for a circular graph voter model

We will consider a voter model where each agent has only left and right neighbors. To avoid boundary effects, we will make use of a circular graph of order n for the network topology. The expression for ρ_i is then

$$\rho_1 = \frac{s_n(t) + s_1(t) + s_2(t)}{3} \tag{5.2.17}$$

$$\rho_i = \frac{s_{i-1}(t) + s_i(t) + s_{i+1}(t)}{3}\ ,\ i \in \{2, \ldots, n-1\} \tag{5.2.18}$$

$$\rho_n = \frac{s_{n-1}(t) + s_n(t) + s_1(t)}{3} \tag{5.2.19}$$

where $s_i(t)$ denotes the state of agent i at time t. Following the coarse graining procedure defined in section 5.2.2, we make groups of 3 cells defined as "super-agents". The Boolean state $S_i(t)$ of the super-agent i is defined by :

$$S_i(t) = \begin{cases} 1 & \text{if } s_{3i+1}(t) + s_{3i+2}(t) + s_{3i+3}(t) \geqslant 2 \\ 0 & \text{otherwise} \end{cases} \tag{5.2.20}$$

Following the notations of figure (5.1), we define the coarse-grained transition probability $P(w|xyz)$ as the probability that $S_i(t+1) = 1$ knowing that the values of $S_{i-1}(t)$, $S_i(t)$ and $S_{i+1}(t)$ are respectively x, y and z. These coarse-grained transition probabilities for super-agents may be computed according to (5.2.12) and the resulting probabilistic cellular automaton may be represented using the notation introduced in (5.2.2). For instance, with a noise $\epsilon = 0.01$, we obtain

$$\begin{aligned} p &= (p_7, p_6, p_5, p_4, p_3, p_2, p_1, p_0) \\ &= (0.9512, 0.7898, 0.3836, 0.2108, 0.7838, 0.6111, 0.2116, 0.0501) \end{aligned}$$

The transition function

We compute the values of $P(1|xyz)$, defined by expression (5.2.12), through intensive simulations with the original CA, for different noise values ϵ. Our hope is that the reduced model can also be described as a 1D voter model. Therefore, we look for a transition function $g : \rho \mapsto \mathbb{P}(S_i(t+3) = 1)$ having an expression similar to $f(\rho)$, namly $g(\rho) = (1-2\epsilon_1)\rho+\epsilon_1$, with ρ depending on $S_{i-1}(t), S_i(t)$ and $S_{i+1}(t)$, for the reduced system which is affine as is the transition function f, of the original system (3.2.4). However, we observe that when choosing

$$\rho = \frac{S_{i-1}(t) + S_i(t) + S_{i+1}(t)}{3}$$

the function g is not affine. It is therefore useful to consider weights for the values of the block states in the expression of ρ. With the values of $P(1|xyz)$ found by sampling, we determine the values α and ϵ_1 in the reduced transition function with weight ratio:

$$\begin{cases} g(\rho) = (1-2\epsilon_1)\rho + \epsilon_1 \\ \rho = \alpha S_{i-1}(t) + (1-2\alpha)S_i(t) + \alpha S_{i+1}(t) \end{cases} \tag{5.2.21}$$

In figure 5.4 (right part), g is computed for several values of ϵ and evaluated for $\rho \in \{0, \alpha, 2\alpha, 1-2\alpha, 1-\alpha, 1\}$. We can see in figure 5.4 (left part) that, for small values of ϵ, ϵ_1 grows linearly with ϵ (the correlation coefficient is 0.99 for values of $\epsilon \in [0; 0.05]$). Using the least squares method, we obtain

$$\epsilon_1 = 3.667\epsilon + 0.0106 \tag{5.2.22}$$

Intuitively, we might think that the value of ϵ_1 includes additional noise introduced by the reduction. The need to weight state values S_i is due to the fact that the state of a block (or super-agent) at time $t+3$ is more influenced by its own state at time t, than by the state of its 2 neighboring blocks. The interactions between the agents can be represented by a causality graph. Its adjacency matrix is, in the case of a periodic (circular) topology:

$$A = \begin{pmatrix} 1 & 1 & 0 & \dots & \dots & 0 & 1 \\ 1 & 1 & 1 & 0 & & & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ 0 & & & \ddots & \ddots & \ddots & 1 \\ 1 & 0 & \dots & \dots & 0 & 1 & 1 \end{pmatrix}$$

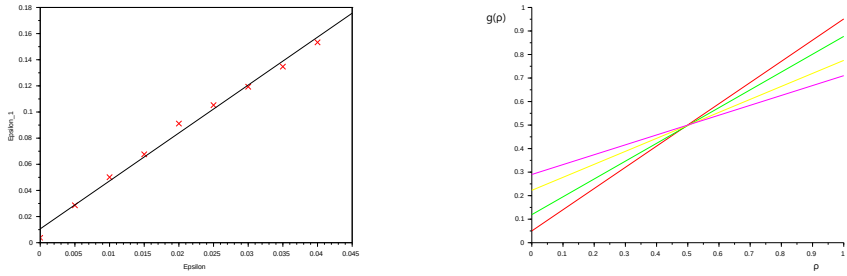

Figure 5.4 – Left : ϵ_1 as a function of ϵ. Right : $g(\rho)$ with different noises : $\epsilon = 0$(red curve), 0.03 (green curve), 0.07 (yellow curve) and $0,1$ (purple curve), values calculated for $\rho \in \{0, \alpha, 2\alpha, 1-2\alpha, 1-\alpha, 1\}$

The number of causal paths between 2 agents after 1, 2 or 3 iterations is then given by the coefficients of the matrix $B = A + A^2 + A^3$. We have:

$$B = \begin{pmatrix} C & D & 0_{3,3} & & & 0_{3,3} & D \\ D & C & D & 0_{3,3} & & & 0_{3,3} \\ 0_{3,3} & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & 0_{3,3} \\ \vdots & & \ddots & \ddots & D & C & D \\ D & O_{3,3} & \ddots & \ddots & \ddots & D & C \end{pmatrix}$$

with

$$C = \begin{pmatrix} 1 & 4 & 9 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix}, \; D = \begin{pmatrix} 9 & 11 & 9 \\ 4 & 9 & 11 \\ 1 & 4 & 9 \end{pmatrix}, \; 0_{3,3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

We may therefore define a parameter b to characterize the influence of the vote of a block of cells at time t on the vote of the same block at time $t+3$ by considering causal paths between agents. We thus consider the ratio of the number of causal paths between the agents in the block to the number of causal paths that exist between the agents in the block and all the agents that will infuence them between instants t and t+3. These numbers are the coefficients of matrices B and C, so we have:

$$b = \frac{\sum d_{i,j}}{\sum c_{i,j} + 2\sum d_{i,j}} = \frac{67}{107} \approx 0.626$$

Similarly, we may define the corresponding influence parameter a for the vote of the two neighbouring blocks at time t on the vote of the central block at time $t+3$:

$$a = \frac{\sum c_{i,j}}{\sum c_{i,j} + 2\sum d_{i,j}} = \frac{20}{107} \approx 0.187$$

Note that these weighting parameters are derived directly from the interaction graph and do not require sampling to be estimated, contrarily to the estimation of the probabilities in the coarse-grained configuration space by simulations.
For all considered values of ϵ, we obtained by sampling for α a mean value of 0.182 and a standard deviation of 0.0028. It may be noticed that these values are very close to the value of a obtained with the matrix B only. Therefore we conjecture that the weights computed from the adjacency matrix B (hence the value of α) give an accurate approximation of the "real" weights which could be computed from the estimation of the coarse-grained configuration probabilities through an intensive simulation effort. For the rest of the simulations in the chapter, we will choose $\alpha = 0.182$.

5.2.4 Coarse-grained dynamics error analysis

Delayed mutual information

To measure the error between the exact and approximated supercell dynamics, we will compute a metric based on the mutual information between these supercells. Roughly speaking, we will consider that the proposed coarse-grained reduced model described in section 5.2.3 is a good approximation of the real system, when the influence of the vote of a block at some time, on the votes of its neighbouring blocks and itself at subsequent times, are similar, both for the reduced and the exact models.

Influence of a block on itself and its neighboring blocks at subsequent times

For the exact block dynamics, we consider the random variables

$$X_i(t) = S_i(t) \tag{5.2.23}$$

where $S_i(t)$ is the supercell state for block i at time t such as defined in equation (5.2.20). We compute the mutual information $w_{i,i}(t,3)$, $w_{i,i+1}(t,3)$ and $w_{i,i+2}(t,3)$ which evaluate the influence of block i at time t, on the state at time $t+3$ of the three consecutive blocks i, $i+1$ and $i+2$ in a steady state.
For the reduced block dynamics, we consider the random variables $Y_i(t)$ which denotes the state of block i at time t computed with the simplified dynamics, as detailed in subsection 5.2.3. For different noise values, we compute the values of $w_{i,i+k}(t,3)$ and $w'_{i,i+k}(t,3)$ (with $k = 0$, 1 and 2) which are the mutual information between the

blocks i and $i+k$ respectively for the exact system and the reduced system. To compute these values, we need to know the different probabilities of the laws of random variables $Y_i(t)$ and $Y_{i+k}(t+3)$ and their joint law. These different probabilities p are obtained by sampling. As the sample size is 10^5 (the number of simulation runs), according to the central limit theorem, the obtained values are with a precision of 0.003 at risk 5% i.e. for each probability p, the estimate p' obtained by sampling verifies $\mathbb{P}(p \in [p'-0.003; p'+0.003]) \geqslant 0.95$. The values calculated for different noises are in the table 5.1.

Table 5.1 – Mutuel information between neighbor blocks.

Noises ϵ	0.01	0.02	0.03	0.04	0.05
$w_{i,i}(t,3)$	0.36	0.24	0.17	0.13	0.10
$w'_{i,i}(t,3)$	0.39	0.29	0.22	0.16	0.13
$w_{i,i+1}(t,3)$	0.15	0.08	0.05	0.031	0.02
$w'_{i,i+1}(t,3)$	0.13	0.07	0.04	0.027	0.02
$w_{i,i+2}(t,3)$	0.03	0.01	0.004	0.002	0.001
$w'_{i,i+2}(t,3)$	0.03	0.008	0.003	0.001	0.0004

The influences between the blocks are consistent within 10% for the simplified and exact models. We can deduce that the proposed reduction offers a coarse-grained model that is a good approximation of the original model. It should be noticed that the relative difference between the values of the mutual information of the 2 models increases with the noise. However, in the case of a very important noise, the mutual information decreases rapidly with the distance and the influence of an agent on the behavior of agent outside its own block may be very low.

5.3 Simplification in the case of a scale free graph

5.3.1 Communities of the graph

Now, we consider the voter model where social interaction are represented by a scale free graph. The dynamics of this complex system is the same as presented in section 5.2.3. We will use the community partition determined in section 4.4 (see figure 5.5) to do a simplification of the system. Each community will be considered as a super cell. The dynamics of the simplified system is described in the following section

Projection using the communities

We simplify the model with the communities of figure 5.5, considering a complete graph where the nodes are the communities, see the figure 5.6.

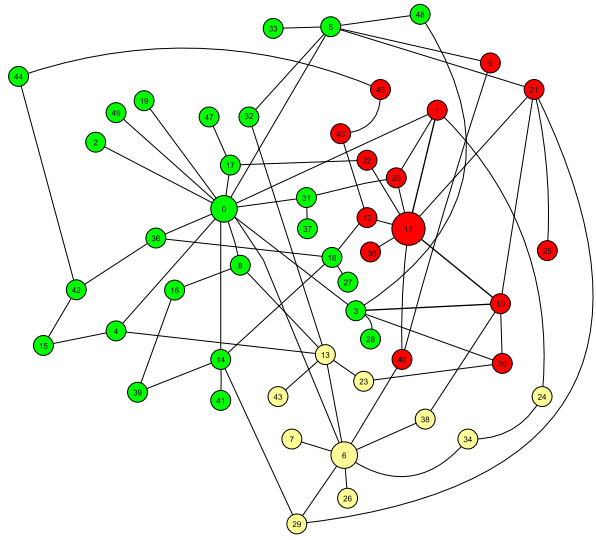

Figure 5.5 – Partition of a scale free graph with 3 communities denoted by the colors.

We have a model with 3 nodes, and we have the states are $x_1x_2x_3$ with

$$x_i = \begin{cases} 1 & \text{if the community } i \text{ votes 1 in majority} \\ 0 & \text{if the community i votes 0 in majority} \end{cases} \tag{5.3.1}$$

Let $X_i(t)$, for $i = 1, 2$ or 3, be the random variable which is equal to the state of the community i.
We note $p_{y_1y_2y_3|x_1x_2x_3}$ the conditionnal probability of the state of the system at time $t + \tau$ knowing the state of the system at time t, i.e.

$$p_{y_1y_2y3|x_1x_2x_3} = P(X_{1:3}(t+\tau) = y_{1:3} \mid X_{1:3}(t) = x_{1:3}) \tag{5.3.2}$$

where:

$$X_{1:3}(t) = \big(X_1(t), X_2(t), X_3(t)\big)$$

$$y_{1:3} = (y_1, y_2, y_3) \text{ and } x_{1:3} = (x_1, x_2, x_3)$$

We take as value of τ, the radius of the graph, which 3 in our example 5.5. We saw in section 4.4.2 that a partition with 3 communities was well suited to our example. So our reduced model is composed of 3 super cells. The transition matrix of this model

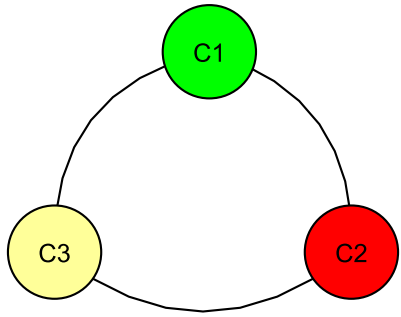

Figure 5.6 – Graph where the nodes are the communities

is

$$P = \begin{pmatrix} p_{000|000} & p_{000|001} & \cdots & p_{000|111} \\ p_{001|000} & p_{001|001} & p_{000|000} & p_{001|111} \\ \vdots & & \ddots & \vdots \\ p_{111|000} & p_{111|001} & \cdots & p_{111|111} \end{pmatrix}$$

We compute the values $p_{y_1y_2y_3|x_1x_2x_3}$ by simulation. For this purpose we count

- the number of times that we go to the state $x_1x_2x_3$, for any $X_1,\ X_2,\ X_3$
- the number of times $n_{y_1y_2y_3|x_1x_2x_3}$ that the transition between these two states, is observed.

Thus we have :

$$p_{y_1y_2y_3|x_1x_2x_3} = \frac{n_{y_1y_2y_3|x_1x_2x_3}}{n_{x_1x_2x_3}} \tag{5.3.3}$$

With 10^7 transitions, we have obtained the transition probability matrix :

$$P = \begin{pmatrix} 0.981 & 0.0035 & 0.0064 & 0.0004 & 0.0071 & 0.0004 & 0.0002 & 0.0006 \\ 0.1908 & 0.4898 & 0.0273 & 0.1060 & 0.0280 & 0.1215 & 0.0063 & 0.02998 \\ 0.2328 & 0.01698 & 0.5367 & 0.064 & 0.0308 & 0.0033 & 0.0958 & 0.0191 \\ 0.018 & 0.071 & 0.0711 & 0.5010 & 0.0052 & 0.0312 & 0.0228 & 0.2789 \\ 0.2603 & 0.0218 & 0.0292 & 0.0054 & 0.5131 & 0.0622 & 0.08953 & 0.01816 \\ 0.0189 & 0.084 & 0.0050 & 0.0296 & 0.07253 & 0.5050 & 0.0218 & 0.2622 \\ 0.0270 & 0.0057 & 0.1117 & 0.0270 & 0.1060 & 0.0220 & 0.5230 & 0.17730 \\ 0.0007 & 0.0004 & 0.0004 & 0.0054 & 0.0004 & 0.0052 & 0.0034 & 0.9844 \end{pmatrix}$$

The τ delayed mutual information between $X_i(t)$ and $X_j(t+\tau)$ is defined by

$$I(i,j) = \sum_{x=0}^{1}\sum_{y=0}^{1} p_{xy}\log_2\left(\frac{p_{xy}}{p_x p_y}\right) \tag{5.3.4}$$

With $\forall(i,j) \in \{1,2,3\}$
$p_{xy} = \mathbb{P}(X_i(t) = x \cap X_j(t+\tau) = y) \quad p_x = \mathbb{P}(X_i(t) = x) \quad p_y = \mathbb{P}(X_i(t+\tau) = y)$
We have
$p_x = \sum_{x_1,x_2,x_3|x_i=x} p_{x_1x_2x_3(t)}$, $p_y = \sum_{y_1,y_2,y_3|y_j=y} p_{y_1y_2y_3}(t+\tau)$ and

$$p_{xy} = \sum_{y_1,y_2,y_3|y_j=y}\ \sum_{x_1,x_2,x_3|x_i=x} p_{y_1y_2y_3|x_1x_2x_3}p_x \tag{5.3.5}$$

If we don't force the state of any nodes, we assume that the values of x and y are uniformly distributed, so we take $p_x = p_y = \frac{1}{8}$.
With these transition probabilities and probability distributions, we may compute the $\tau-$mutual information matrix $M' = \big(I(i,j)\big)_{1\leqslant i,j\leqslant 3}$ between any pair of super agent. In a stationary regime with the delay $\tau = 3$ and a noise $\epsilon = 0.001$. We obtain:

$$M_1 = \begin{pmatrix} 0.792 & 0.731 & 0.731 \\ 0.730 & 0.744 & 0.709 \\ 0.728 & 0.708 & 0.735 \end{pmatrix}$$

We can compare these values with theses of the delayed-mutual information between the state of the communities in the case of the original system. We used the same delay and the same noise and computed the latter values by sampling with a sample size $N = 10^7$.

$$M = \begin{pmatrix} 0.826 & 0.747 & 0.750 \\ 0.747 & 0.796 & 0.722 \\ 0.746 & 0.720 & 0.785 \end{pmatrix}$$

According to the central limit theorem, with a sample of size $N = 10^5$, we obtain values approximated to within 0.03 at a risk of 5%. Thus, We can see that the delayed influence between communities are similar in the case of the real system and the reduced system.

5.4 Conclusions

Our goal was to explore the possibility to simplify (stochastic) complex systems on graphs by reducing the number of degrees of freedom, while keeping the essential dynamical features. We believe that the originality of the proposed method is that it can be applied for any projection function and any probabilistic or deterministic CAs. In

our theoretical framework an important aspect to correctly build the simplified system is to have hypotheses on the probability distribution of all possible configurations. In the case of the voter model, where we determined the reduced system from a statistical analysis of the real one, we observed that coarse grained agents may have to be defined with a weighted average of the real ones. Anyway, the examples we showed in this chapter suggest that system reduction is a possible avenue, in particular with the objective of controllability. However, numerous issues need to be investigated: the evaluation of the information loss in the reduction, the comparisons of the stationary probability distributions, etc. Another question is how to best define the blocks or super-agents for the system reduction. When applying the reduction on a dynamical system defined on a graph, a natural way could be to consider the partitions associated with graph communities.

6 | Conclusion and prospects

Conclusion

In this thesis, we set out to show that information theory offers tools for studying complex systems, and that these tools provide an answer to the problem of reachability and observability in control theory. To illustrate the approach, we used a voter model which is defined by fairly simple dynamics but still is representative of many complex systems.

First, we defined delayed information and multi-information inspired by the mutual transfer in Schreiber [96]. We have shown that these quantities make it possible to quantify the influence and determine the causal relationships between agent's state (vote) over time. Multi-information allowed us to determine which agents had the most influence on the system, and we were able to control the system while limiting the cost of control.

Next, we saw in the case of a system with an unknown topology that we were able to reconstruct the causal graph by observing its behavior over time. Besides, we also showed that the mutual information calculated over a sliding time window could be used to detect changes in the topology that had an influence on the state of the system. More precisely, the 1-delayed mutual information computed on a sliding-window was used to identify the possible changes of connectivity. We saw that this quantity allows us to detect whether an edge is added or removed between two vertices of low degree, or between a hub and a vertex of low degree. Between two hubs this method is not effective, but the actual presence or absence of a link between them does not affect much the global behavior of the system. Detecting structural changes is important as it can provide an early warning of tipping points. We expect that our approach can be applied to many other complex systems beyond the voter model.

Finally, we have used information theory to propose a method for decomposing the system into a partition of communities of influence subsets of agents strongly linked by dynamic. This partition is then used to simplify large complex systems. Our goal was to explore the possibility to simplify stochastic complex systems on graphs by reducing the number of degrees of freedom, while keeping the essential dynamical features. An important aspect to correctly build the simplified system is the probability distribution of all possible configurations in the original model. In the case of the voter model, where we determined the reduced system from a statistical analysis of the real one, we observed that coarse grained agents may have to be defined with a

weighted average of the real ones. The corresponding weights may be obtained by using information on the graph topology.

We believe that the originality of the proposed reduction method is that it can be applied for any projection function and any probabilistic CAs. We first looked at a cellular automaton that follows Wolfram's rules, and then at a 1D voting system on a ring (1D topology). Mutual information was used to check that the simplified model behaves like the original one and enabled us to validate the reduction method. This method was then generalised to any graph, using communities defined from mutual information and identified as super agents in the reduced model. The investigated examples suggest that system reduction based on mutual information analysis is a possible avenue for the reduction of large scale systems, in particular with a control objective in mind.

Generally speaking we may conclude that information theory is well suited to solve some control problems for complex systems. It allows us to carry out a structural analysis and provides indicators that can facilitate the investigation of these control problems.

Prospects

The work carried out in this thesis can be extended in various directions.

Firstly, we would like to apply the results obtained on the voter model to the world3 model. The World3 model is a system dynamics model for computer simulation of interactions between population, industrial growth, food production and limits in the ecosystems of the earth [66, 43]. The dynamics of this model are defined by several differential equations whose functions are the various parameters of the model as a function of time. From these differential equations, it is not easy to assess the influence of each parameter on the others. Using delayed mutual information, we would like to quantify these influences in order to determine which actions would be most effective in achieving the desired objectives. For example, if we want to reduce pollution, we can identify which parameters need to be modified to achieve the desired effect quickly. The evaluation of causality should also enable us to propose a model reduction that will simplify our study.

Regarding model reduction, numerous issues still need to be investigated such as the evaluation of the information loss in the reduction or the comparisons of the stationary probability distributions. Another question is how to best define the blocks or super-agents for the system reduction. In our approach, these blocks are communities defined with respect to the higher multi-information (seeds) and mutual information. However, there are many other ways to define communities and a comparative study is still needed.

We have shown that it is possible to detect changes in the topology or in the dynamical behavior of a system over time using information-theoretic tools. This could be the basis for the definition of early warnings. For instance, economic models and changes in the topology or in the agents' state instantaneous behavior could be used to alert us about the risk of a looming economic crisis.

Another area of extension concerns the estimation problem. In many practical situations, we are not able to measure the whole state at all times. In such situations, hidden Markov chains could be used to recover the missing data from the available measurements. Information theoretic tools such as those presented in this thesis could be used to test the feasability (observability) or help in the estimation.

Bibliography

[1] Tatsuya Akutsu, Morihiro Hayashida, and Takeyuki Tamura. Algorithms for inference, analysis and control of boolean networks. In *Algebraic Biology: Third International Conference, AB 2008, Castle of Hagenberg, Austria, July 31-August 2, 2008 Proceedings 3*, pages 1–15. Springer, 2008.

[2] Pierre-Olivier Amblard and Olivier JJ Michel. On directed information theory and granger causality graphs. *Journal of computational neuroscience*, 30(1):7–16, 2011.

[3] Pierre-Olivier Amblard and Olivier JJ Michel. The relation between granger causality and directed information theory: A review. *Entropy*, 15(1):113–143, 2013.

[4] Athanasios C Antoulas. *Approximation of large-scale dynamical systems*. SIAM, 2005.

[5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[6] Albert-László Barabási, Réka Albert, and Hawoong Jeong. Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A: statistical mechanics and its applications*, 281(1-4):69–77, 2000.

[7] Guillaume Becq, Pierre-Olivier Amblard, Paul Salin, and Jean-Christophe Comte. Détection dynamique de graphes appliquée à l'analyse de signaux électroencéphalographiques issus du sommeil de rat. In *GRETSI 2017-XXVIème Colloque francophone de traitement du signal et des images*, 2017.

[8] Abdelghani Bellouquid, Elena De Angelis, and LUISA FERMO. Towards the modeling of vehicular traffic as a complex system: A kinetic theory approach. *Mathematical models and methods in applied sciences*, 22(supp01):1140003, 2012.

[9] Peter Benner, Jing-Rebecca Li, and Thilo Penzl. Numerical solution of large-scale lyapunov equations, riccati equations, and linear-quadratic optimal control problems. *Numerical Linear Algebra with Applications*, 15(9):755–777, 2008.

[10] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[11] S Boccaletti, M Ivanchenko, V Latora, A Pluchino, and A Rapisarda. Detecting complex network modularity by dynamical clustering. *Physical Review E*, 75(4):045102, 2007.

[12] Béla Bollobás and Oliver Riordan. The diameter of a scale-free random graph. *Combinatorica*, 24(1):5–34, 2004.

[13] Béla Bollobás and Oliver M Riordan. Mathematical results on scale-free random graphs. *Handbook of graphs and networks: from the genome to the internet*, pages 1–34, 2003.

[14] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Gorke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On modularity clustering. *IEEE transactions on knowledge and data engineering*, 20(2):172–188, 2007.

[15] Markus Brede. How does active participation affect consensus: Adaptive network model of opinion dynamics and influence maximizing rewiring. *Complexity*, 2019, 2019.

[16] Markus Brede, Valerio Restocchi, and Sebastian Stein. Resisting influence: how the strength of predispositions to resist control can change strategies for optimal opinion control in the voter model. *Frontiers in Robotics and AI*, 5:34, 2018.

[17] Timoteo Carletti, Duccio Fanelli, and Renaud Lambiotte. Random walks and community detection in hypergraphs. *Journal of Physics: Complexity*, 2(1):015011, 2021.

[18] Claudio Castellano, Miguel A Muñoz, and Romualdo Pastor-Satorras. Nonlinear q-voter model. *Physical Review E*, 80(4):041129, 2009.

[19] Abraham C-L Chian. *Complex systems approach to economic dynamics*, volume 592. Springer Science & Business Media, 2007.

[20] Gregor Chliamovitch. *Information theory and maximum entropy principles in non-equilibrium statistical physics*. PhD thesis, Université de Genève, 2017.

[21] Fabien Claveau. *Contribution à l'analyse et la commande structurée des grands systèmes*. PhD thesis, Ecole Centrale de Nantes (ECN); Université de Nantes, 2005.

[22] Christian Commault and Jean-Michel Dion. Input addition and leader selection for the controllability of graph-based systems. *Automatica*, 49(11):3322–3328, 2013.

[23] Christian Commault and Jean-Michel Dion. The minimal controllability problem for structured systems. 2014.

[24] Pedro Costa and Fernando De Melo. Coarse graining of partitioned cellular automata. *arXiv preprint arXiv:1905.10391*, 2019.

[25] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[26] Brigitte d'Andréa Novel and Michel Cohen De Lara. *Commande linéaire des systèmes dynamiques*. Presses des MINES, 2000.

[27] Jean-Charles Delvenne, Michael T Schaub, Sophia N Yaliraki, and Mauricio Barahona. The stability of a graph partition: A dynamics-based framework for community detection. *Dynamics On and Of Complex Networks, Volume 2: Applications to Time-Varying Dynamical Systems*, pages 221–242, 2013.

[28] Jean-Michel Dion, Christian Commault, and Jacob Van der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39(7):1125–1144, 2003.

[29] Dirk Drasdo. Coarse graining in simulated cell populations. *Advances in Complex Systems*, 8(02n03):319–363, 2005.

[30] Mathilde Jochaud Du Plessix. *Analyse du modèle World3: sensibilité, dynamique, et pistes d'évolution*. PhD thesis, Insa Lyon, 2019.

[31] Paul Erdős, Alfréd Rényi, et al. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[32] David P Feldman and Jim Crutchfield. A survey of complexity measures. *Santa Fe Institute, USA*, 11, 1998.

[33] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3-5):75–174, 2010.

[34] Santo Fortunato and Marc Barthelemy. Resolution limit in community detection. *Proceedings of the national academy of sciences*, 104(1):36–41, 2007.

[35] John Foster. From simplistic to complex systems in economics. *Cambridge Journal of Economics*, 29(6):873–892, 2005.

[36] Serge Galam and Frans Jacobs. The role of inflexible minorities in the breaking of democratic opinion dynamics. *Physica A: Statistical Mechanics and its Applications*, 381:366–376, 2007.

[37] Jianxi Gao, Yang-Yu Liu, Raissa M D'souza, and Albert-László Barabási. Target control of complex networks. *Nature communications*, 5(1):5415, 2014.

[38] Sayan Ghosh, Mahantesh Halappanavar, Antonino Tumeo, Ananth Kalyanaraman, Hao Lu, Daniel Chavarria-Miranda, Arif Khan, and Assefaw Gebremedhin. Distributed louvain algorithm for graph community detection. In *2018 IEEE international parallel and distributed processing symposium (IPDPS)*, pages 885–895. IEEE, 2018.

[39] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[40] Clive WJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, pages 424–438, 1969.

[41] Clive WJ Granger. Some recent development in a concept of causality. *Journal of econometrics*, 39(1-2):199–211, 1988.

[42] Hendrik Hache, Hans Lehrach, and Ralf Herwig. Reverse engineering of gene regulatory networks: a comparative study. *EURASIP Journal on Bioinformatics and Systems Biology*, 2009:1–12, 2009.

[43] Gaya Herrington. Update to limits to growth: Comparing the world3 model with empirical data. *Journal of Industrial Ecology*, 25(3):614–626, 2021.

[44] Suriana Ismail and Roslan Ismail. Modularity approach for community detection in complex networks. In *Proceedings of the 11th international conference on ubiquitous information management and communication*, pages 1–6, 2017.

[45] Navot Israeli and Nigel Goldenfeld. Coarse-graining of cellular automata, emergence, and the predictability of complex systems. *Physical Review E*, 73(2):026203, 2006.

[46] Peter D Johnson Jr, Greg A Harris, and DC Hankerson. *Introduction to information theory and data compression*. CRC press, 2003.

[47] R. E. Kalman. Mathematical description of linear dynamical systems. *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, 1(2):152–192, 1963.

[48] Rudolf E Kalman. On the general theory of control systems. In *Proceedings First International Conference on Automatic Control, Moscow, USSR*, pages 481–492, 1960.

[49] Rushed Kanawati. Seed-centric approaches for community detection in complex networks. In *International Conference on Social Computing and Social Media*, pages 197–208. Springer, 2014.

[50] Kunihiko Kaneko. *Life: an introduction to complex systems biology*. Springer, 2006.

[51] Brian Karrer, Elizaveta Levina, and Mark EJ Newman. Robustness of community structure in networks. *Physical review E*, 77(4):046119, 2008.

[52] Chris J Kuhlman, VS Anil Kumar, and SS Ravi. Controlling opinion propagation in online networks. *Computer Networks*, 57(10):2121–2132, 2013.

[53] James Ladyman, James Lambert, and Karoline Wiesner. What is a complex system? *European Journal for Philosophy of Science*, 3(1):33–67, 2013.

[54] Renaud Lambiotte, Jean-Charles Delvenne, and Mauricio Barahona. Random walks, markov processes and the multiscale modular organization of complex networks. *IEEE Transactions on Network Science and Engineering*, 1(2):76–90, 2014.

[55] Rong-Hua Li, Jeffrey Xu Yu, Xin Huang, and Hong Cheng. Random-walk domination in large graphs. In *2014 IEEE 30th International Conference on Data Engineering*, pages 736–747. IEEE, 2014.

[56] Ching Tai Lin. Structural controllability. *Automatic Control, IEEE Transactions on*, 19(3):201–208, 1974.

[57] Jiamou Liu and Ziheng Wei. Community detection based on graph dynamical systems with asynchronous runs. In *2014 Second International Symposium on Computing and Networking*, pages 463–469. IEEE, 2014.

[58] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabasi. Controllability of complex networks. *Nature*, 473:167, 2011.

[59] Yang-Yu Liu and Albert-Laszló Barabási. Control principles of complex networks. *arXiv preprint arXiv:1508.05384*, 2015.

[60] Yang-Yu Liu and Albert-László Barabási. Control principles of complex systems. *Reviews of Modern Physics*, 88(3):035006, 2016.

[61] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. Observability of complex systems. *Proceedings of the National Academy of Sciences*, 110(7):2460–2465, 2013.

[62] Mejda Mansouri, Latifa Boutat-Baddas, Mohamed Darouach, and Hassani Messaoud. Decentralized observers for a class of large-scale singular systems via lmi. *International Journal of Intelligent Computing and Cybernetics*, 6(2):158–181, 2013.

[63] Naoki Masuda. Opinion control in complex networks. *New Journal of Physics*, 17(3):033031, 2015.

[64] Ueli M Maurer. The role of information theory in cryptography. In *Fourth IMA Conference on Cryptography and Coding*, pages 49–71. Citeseer, 1993.

[65] Charalampos Mavroforakis, Michael Mathioudakis, and Aristides Gionis. Absorbing random-walk centrality: Theory and algorithms. In *2015 IEEE International Conference on Data Mining*, pages 901–906. IEEE, 2015.

[66] Dennis Meadows and Jorgan Randers. *The limits to growth: the 30-year update*. Routledge, 2012.

[67] Dennis L Meadows, William W Behrens, Donella H Meadows, Roger F Naill, Jørgen Randers, and Erich Zahn. *Dynamics of growth in a finite world*. Wright-Allen Press Cambridge, MA, 1974.

[68] Stanley Milgram. The small world problem. *Psychology today*, 2(1):60–67, 1967.

[69] Mauro Mobilia. Does a single zealot affect an infinite group of voters? *Physical review letters*, 91(2):028701, 2003.

[70] Mauro Mobilia, Anna Petersen, and Sidney Redner. On the role of zealotry in the voter model. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(08):P08029, 2007.

[71] Flaviano Morone and Hernán A Makse. Influence maximization in complex networks through optimal percolation. *Nature*, 524(7563):65, 2015.

[72] Mark Newman. *Networks*. Oxford university press, 2018.

[73] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[74] Mark EJ Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104, 2006.

[75] Mark EJ Newman. Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582, 2006.

[76] Mark EJ Newman. Complex systems: A survey. *arXiv preprint arXiv:1112.1440*, 2011.

[77] Mark EJ Newman, Duncan J Watts, and Steven H Strogatz. Random graph models of social networks. *Proceedings of the National Academy of Sciences*, 99(suppl 1):2566–2572, 2002.

[78] Piotr Nyczka and Katarzyna Sznajd-Weron. Anticonformity or independence?—insights from statistical physics. *Journal of Statistical Physics*, 151(1):174–202, 2013.

[79] Piotr Nyczka, Katarzyna Sznajd-Weron, and Jerzy Cisło. Phase transitions in the q-voter model with two types of stochastic driving. *Physical Review E*, 86(1):011105, 2012.

[80] Carlos Ocampo-Martínez, Samuele Bovo, and Vicenç Puig. Partitioning approach oriented to the decentralised predictive control of large-scale systems. *Journal of Process Control*, 21(5):775–786, 2011.

[81] Symeon Papadopoulos, Yiannis Kompatsiaris, Athena Vakali, and Ploutarchos Spyridonos. Community detection in social media. *Data Mining and Knowledge Discovery*, 24(3):515–554, 2012.

[82] Fabio Pasqualetti, Sandro Zampieri, and Francesco Bullo. Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1):40–52, 2014.

[83] Aurelio Patelli, Andrea Gabrielli, and Giulio Cimini. Generalized markov stability of network communities. *Physical Review E*, 101(5):052301, 2020.

[84] J. Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2009.

[85] Judea Pearl. Causal inference. *Causality: objectives and assessment*, pages 39–58, 2010.

[86] Sen Pei, Jiannan Wang, Flaviano Morone, and Hernán A Makse. Influencer identification in dynamical complex systems. *Journal of complex networks*, 8(2):cnz029, 2020.

[87] Derek de Solla Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American society for Information science*, 27(5):292–306, 1976.

[88] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. Defining and identifying communities in networks. *Proceedings of the national academy of sciences*, 101(9):2658–2663, 2004.

[89] Ranaivo Mahaleo Razakanirina and Bastien Chopard. Risk analysis and controllability of credit market. *ESAIM: Proceedings and Surveys*, 49:91–101, 2015.

[90] Pedro Retortillo, Margarita Mediavilla, Luis Javier Miguel, and C Castro. An attempt to automate the analysis of complex system dynamics models: an example of world 3. In *Proceedings of the 26th International Conference of the System Dynamics Society, Athens, Greece*, 2008.

[91] Fazlollah M Reza. *An introduction to information theory*. Courier Corporation, 1994.

[92] David Anthony Richardson. *Communication and Information Theory, Cryptography and Applications*. PhD thesis, University of Calgary, Department of Mathematics & Statistics, 2006.

[93] Olivier Rioul. *Théorie de l'information et du codage*. Lavoisier Paris, 2007.

[94] Sandip Roy and Mengran Xue. Controllability-gramian submatrices for a network consensus model. *arXiv preprint arXiv:1903.09125*, 2019.

[95] Julien Salotti, Serge Fenet, Romain Billot, Nour-Eddin El Faouzi, and Christine Solnon. Comparison of traffic forecasting methods in urban and suburban context. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 846–853. IEEE, 2018.

[96] Thomas Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000.

[97] Anil K Seth, Adam B Barrett, and Lionel Barnett. Granger causality analysis in neuroscience and neuroimaging. *Journal of Neuroscience*, 35(8):3293–3297, 2015.

[98] Claude Elwood Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

[99] DD Šiljak and MB Vukčević. Large-scale systems: Stability, complexity, reliability. *Journal of the Franklin Institute*, 301(1-2):49–69, 1976.

[100] Dragoslav D Siljak. *Decentralized control of complex systems*. Courier Corporation, 2011.

[101] Dragoslav D Šiljak and AI Zečević. Control of large-scale systems: Beyond decentralized feedback. *Annual Reviews in Control*, 29(2):169–179, 2005.

[102] Roland Somogyi, Stefanie Fuhrman, Manor Askenazi, and Andy Wuensche. The gene expression matrix: towards the extraction of genetic network architectures. *Nonlinear Analysis*, 30(3):1815–1824, 1997.

[103] Daniel F Spulber and Christopher S Yoo. On the regulation of networks as complex systems: A graph theory approach. *Nw. UL Rev.*, 99:1687, 2004.

[104] Patrick A Stokes. *Fundamental problems in Granger causality analysis of neuroscience data*. PhD thesis, Massachusetts Institute of Technology, 2015.

[105] Patrick Suppes. Probabilistic causality in space and time. In *Causation, Chance and Credence*, pages 135–151. Springer, 1988.

[106] Pierre-Alain Toupance, Bastien Chopard, and Laurent Lefevre. Identification of complex network topologies through delayed mutual information. *IFAC-PapersOnLine*, 53(2):1019–1024, 2020.

[107] Henk J Van Waarde, M Kanat Camlibel, and Harry L Trentelman. A distance-based approach to strong target control of dynamical networks. *IEEE Transactions on Automatic Control*, 62(12):6266–6277, 2017.

[108] Amirkhosro Vosughi, Charles Johnson, Mengran Xue, Sandip Roy, and Sean Warnick. Target control and source estimation metrics for dynamical networks. *Automatica*, 100:412–416, 2019.

[109] Duncan J Watts and Peter Sheridan Dodds. Influentials, networks, and public opinion formation. *Journal of consumer research*, 34(4):441–458, 2007.

[110] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[111] Nobert Wiener. What is information theory. *IRE Transactions on Information Theory*, 2(2):48, 1956.

[112] Norbert Wiener. *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press, 2019.

[113] Andreas Wilmer, Marc de Lussanet, and Markus Lappe. Time-delayed mutual information of the phase as a measure of functional connectivity. *PloS one*, 7(9), 2012.

[114] Stephen Wolfram. Cellular automata. *Los Alamos Science*, pages 09–01, 1983.

[115] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of modern physics*, 55(3):601, 1983.

[116] Andrew Y Wu, Michael Garland, and Jiawei Han. Mining scale-free networks using geodesic clustering. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 719–724. ACM, 2004.

[117] Bo Yang, Dayou Liu, and Jiming Liu. Discovering communities from social networks: Methodologies and applications. *Handbook of social network technologies and applications*, pages 331–346, 2010.

[118] Qin Zhang and Shichao Geng. Dynamic uncertain causality graph applied to dynamic fault diagnoses of large and complex systems. *IEEE Transactions on Reliability*, 64(3):910–927, 2015.