UCL
Université catholique de Louvain

École polytechnique de Louvain (EPL)

Open Multi-Agent Systems with Fixed-Sized and Possibly Not Complete Topologies

Dissertation presented by
Alexandre OLIKIER

for obtaining the Master's degree in
Mathematical Engineering

Supervisors
Paolo FRASCA (CNRS), Julien HENDRICKX (UCL)

Reader
Jean-Charles DELVENNE (UCL)

Academic year 2017-2018

Abstract

Multi-agent systems are systems composed of intelligent agents interacting with one another. Such systems are commonly represented by a graph in which each agent is a node holding a randomly distributed value and each interaction is an edge. They have proved to be a powerful tool in various applications such as opinion dynamic modelling. However, in many applications, especially when the size of the system is large, agents are likely to leave or enter the system. In other words, at each time step, nodes and edges can be added or deleted in the graph representing the system. Such systems, called open multi-agent systems, have been recently introduced and are not yet well understood. More precisely, they have been analyzed only in the simple case where the graph representing the system is complete. The goal of this master thesis is to investigate other graph structures with the only constraint that the number of nodes does not vary with time, *i.e.*, the arrival of an agent always coincides with the departure of another one. For instance, the behaviour of bipartite graphs is studied. A rule between the structure of a system and the variance of the values held by the agent is also provided. As an application, the PageRank problem is revisited using an open decentralized method.

KEYWORDS: open multi-agent systems, multi-agent systems, distributed artificial intelligence, PageRank, gossip, IoT, agent, decentralized method.

Acknowledgments

First of all, I wish to express my sincere gratitude to my supervisor at the *Université catholique de Louvain*, Prof. Julien Hendrickx, for his continuous support of my master thesis and related research, for his patience, motivation, and immense knowledge. He has always been available and sensitive to my requests, especially his help in finding an international collaboration for me. His guidance helped me throughout.

I sincerely want to thank my second supervisor at the *Centre National de la Recherche Scientifique*, Prof. Paolo Frasca, for his advice and the continuous tracking of my work during my two months of internship at the *GIPSA-LAB* in Grenoble. He spent an incredible amount of time with me resolving my issues. More than a supervisor, he really helped me in the contents and structure of this thesis; it would not be what it is without him. Beside his professional qualities, he also introduced me to other members of the *GIPSA-LAB*, concerned about my integration. For all these reasons, I feel very indebted to him.

I am very grateful to Alexander Artikis for kindly sending his article when I asked him. It considerably helped my knowledge about open multi-agent systems.

I also want to thank Gauthier Muguerza, Guillaume Olikier and Peter Shaw for their advice and the proofreading of my work. Last but not least, I thank my family and friends for their help and support.

Contents

Notation

$\overline{x}(k)$	The mean of vector $x \in \mathbb{R}^n$ at time k. If x is of size n, $\overline{x}(k) = \frac{1}{n}\sum_{i=1}^{n} x_i(k)$.
$\overline{x^2}(k)$	The mean square value of vector $x \in \mathbb{R}^n$ at time k. If x is of size n, $\overline{x^2}(k) = \frac{1}{n}\sum_{i=1}^{n} x_i^2(k)$.
$\mathrm{Var}(x(k))$	The variance of vector $x \in \mathbb{R}^n$ at time k. If x is of size n, $\mathrm{Var}(x(k)) = \overline{x^2}(k) - \overline{x}^2(k)$.
$\mathcal{P}(\rho)$	The Poisson process of parameter ρ.
$\mathbb{E}(\overline{x}^2)\|_{\mathrm{eq}}$	The fixed point of the square mean value.
$\mathbb{E}(\overline{x^2})\|_{\mathrm{eq}}$	The fixed point of the mean square value.
$\mathbb{E}(\mathrm{Var}(x))\|_{\mathrm{eq}}$	The fixed point of the variance.
$\mathcal{G}(\mathcal{V}, \mathcal{E})$	A graph $\mathcal{G}$ with $\mathcal{V}$ the set of vertices/agents of $\mathcal{G}$ and $\mathcal{E}$, the set of edges of $\mathcal{G}$.
n	The number of vertices/agents in graph $\mathcal{G}$. n is the cardinal of $\mathcal{V}$
$\mathcal{V}_a, \mathcal{V}_b$	The two partitions of a bipartite graph. $\mathcal{V}_a \cup \mathcal{V}_b = \mathcal{V}$
n_a	The size of $\mathcal{P}_a$.
n_b	The size of $\mathcal{P}_b$.
$\mathcal{O}_i(k)$	The set of out-neighbours of vertex/agent i at time k in a directed graph.
$\mathcal{I}_i(k)$	The set of in-neighbours of vertex/agent i at time k in a directed graph.
P_c	The probability that two agents are connected by an edge in a random graph (Definition 3.3.2).
$\\|\cdot\\|$	The 2-norm of a vector.
$\mathbb{N}$	The set of nonnegative integers.
$\mathbb{N}_0$	The set of positive integers.
p	The probability that a replacement (Definition 1.0.2) occurs in the system at each time step.
The variance of a graph	These few words mean *the variance of the agents of the graph.*

Introduction

Nowadays, computational problems, arising in many real-life situations, are increasingly more complex, requiring ever increasing resources to solve. Consequently, new solving paradigms such as *distributed artificial intelligence* have emerged. Distributed artificial intelligence is the science of distributing, coordinating and predicting the performance of tasks, goals or decisions in a multiple agents environment [A.02]. Since the mid-1970s, it evolved and diversified rapidly [Wei99]. Mutli-agent systems are an example of distributed artificial intelligence. A multi-agent system is a system in which the member agents are developed by different parties and serve different, often competing, interests [AAR16]. It is represented by a graph where each node is an agent and each edge is a channel of communication between two agents. On the one hand, the analysis of multi-agent systems is difficult. Indeed, an important feature of these systems is that the behaviour of an agent cannot be predicted in advance [Hew91] requiring the use of specific probabilistic tools to study their average behaviour. On the other hand, multi-agent systems offer various advantages.

First of all, the interactions among the agents are not mandated by a central controller [A.02], *i.e.*, multi-agent systems do not need a centralized entity which has to deal with all the information contained in the system [BGPS06]. We say that multi-agent systems work in a decentralized way. This is a very important point because it means that when working on a large system or network, decentralized algorithms do not need to know the entire structure of the system. For a network such as the internet, which is made of billions of pages [dK], this property is essential. Moreover, this property allows to solve problems that are very difficult for an individual agent notably because the amount of data is so huge that it is not possible to use a single agent or a single computer [DAW12].

Secondly, decentralized solutions are more robust than centralized ones. Indeed, in a multi-agent system, the risk of failure is dispatched over all the agents while it is concentrated on only one entity in a centralized setting. On top of that, in case of problems, each agent has access to an estimate answer, while in a centralized system the single agent may not be able to provide any solution.

As a matter of fact, multi-agent systems have been used successfully in various applications. In [BHT09], the authors study a model of opinion dynamic using multi-agent systems. In [BHOT05], [OSFM07] and [Mor05], the authors work on an algorithm reaching a consensus between different agents. Each agent holds a value and a number of agents are exchanging their values to form weighted averages with the value possessed by their neighbours. Still in the context of consensus, [KBS06] proposes bounds for the convergence time of these sorts of algorithms for networks with a complete graph structure and also for linear networks.

All these articles concern decentralized methods and/or multi-agent systems, assuming that the structure of the system is frozen. In [GJ12], the authors are working on opinion networks but consider that replacements, *i.e.*, simultaneous arrivals and departures of agents, occur with a given probability. Such multi-agent systems, where replacements are possible, are called *open multi-agent systems*. They share the characteristics presented in the above paragraphs with multi-agent systems, but are able to cope with networks that change their structure. This is a key property because, since decentralized methods are typically used for large problems, there is high probability that agents join the system, leave the system, are replaced or fail. For this reason, being able to react when a modification in the system occurs and being able to adapt its modelling are essential. This point strongly motivates the utility of open multi-agent systems and illustrates one of the limits of multi-agent systems. Given this, we will define an open multi-agent system as a multi-agent system where the agents can leave or enter the system.

However, this sort of systems is quite recent and is not yet well understood from the theoretical point of view. For instance, the analysis conducted in [GJ12] was mostly based on simulations. For this reason, there is a need for further explanations and representation techniques of open multi-agent systems. Through their article, the authors of [HM17] pave the way for this research direction. In [HM17], each agent holds a value. Gossips, which correspond to a pairwise average between two agents, conducting to a consensus occur with a probability $1-p$ while replacements of an agent occur with a probability $p \in [0,1]$. Not only the article provides the steady state equation (Definition 1.0.4) when replacements and gossips occur, but also introduces the notion of fixed point (Definition 1.1.1) and serves as an introduction for Chapter 1. In the first three chapters of this document, we will be working with undirected graphs and the value held by each agent will be a uniformly distributed random variable whilst in Chapter 4, this value will be a number between 0 and 1 and the considered graphs will be directed. *The goal of this master thesis is to broaden results about open multi-agent systems.* To achieve this, different sorts of networks will be considered and a particular application of such systems will be developed.

In Chapter 1, we will extend the research made in [HM17] by studying open multi-agent systems where each agent can communicate with all the others. In this setting, systems are said to be in a complete graph topology. This sort of structure is the most simple one since each agent plays the same role as any other agent and each agent can communicate with any other node of the graph. At each time step, two events are possible with an associated probability: a gossip or a replacement. We will study how the system we are working with behaves in terms of square mean value and variance of the values held by the agents, following different definitions for these two events at each time step. First of all by modifying the number of agents that are implied in a gossip. Then, by studying the consequences when, in a gossip, an agent is more of less influenced by the second one.

Chapter 2 will expand results of the first chapter to a new type of structure, namely bipartite graphs. We will study each partition of this sort of graphs separately, and then we will make a comparison with the complete graph. A study of the fixed point of this sort of structure will also be provided. We will end this chapter by focusing on the particular case of the star graph. We will see that this sort of graph is very interesting since *it is possible to resume the behaviour of the system to the behaviour of a particular agent of this graph.* Through this chapter, we will also see that the variance of the values held by the agents any bipartite graph is always above the one of the complete graph with the same number of agents. This observation will be the motivation for the third chapter.

In Chapter 3, we will explain how the expected variance of the values held by the agents of a graph behaves following the structure of the system we are working with. Indeed, full analyses are only available for complete graphs and for some other rather special structures such as complete bipartite graphs. However, the evolution on non-complete graphs can not be understood by studying the complete graphs. We will first of all verify whether the behaviour observed on complete graphs depends on the vertex-transitivity property. As this property fails to explain the observed behaviour, we will broaden our researches and study the effects of the diameter of a graph on its variance. If this approach is interesting for graphs with a small diameter, it becomes no longer possible to draw a conclusion for graphs with a larger diameter. Finally, we will analyze the eigenvalues of the expected transition matrix (see Section 3.4) associated to the considered system, drawing from the results of [BGPS06] slightly adapted to our problem. *Using* *this method allows us to find some bounds on the variance of a system.* Another statement is that the variance observed on complete graphs behave like a lower bound for the variance the values held by the agents of any other graph.

Chapter 4 will be quite different to the first three. Indeed, we will apply open multi-agent systems to a well-known algorithm, namely the PageRank. The PageRank algorithm was created in the late 1990s [Rob16] and has been subject to many modifications since. The first versions of the PageRank suffered from some issues that have now been fixed. We know from [RFTI15] that it is possible to compute the PageRank of a given graph by using open multi-agent systems and by defining gossip in a certain way that provides convergence towards results of the classical PageRank. In [RFTI15], although the article is more about ergodicity of multi-agent systems, we can also pay attention to the definition of gossips for a decentralized PageRank algorithm. This definition guarantees in average a convergence towards results of the classical PageRank which does not rely on open multi-agent systems. However, a point has not been explored: *What happens when replacements occur in the system,* i.e.*, can we cope with new agents and is it possible to keep the proper functioning announced when only gossips occur?* Through this chapter, we will work in this direction, and we will explain how replacements should be defined. Note that, in this chapter, the value assigned to an agent will no longer follow a given distribution, but will depend on the structure of the considered system. We will also extend the utility of this algorithm by allowing replacement in our graph such that our algorithm compute the PageRank of a graph quite accurately despite some modifications in the structure of the system we are working with. To achieve this we will try different definitions for a replacement, and we will keep the best one. A very important point is that in the scope of this chapter, the definition of a replacement will be different since it will modify the structure of the graph.

Through this document, we will be working on decentralized methods where it is possible to modify the structure of the considered system. For this reason, we will say that we are working with *open* decentralized methods. In each chapter, the technique representations used are able to cope with replacements, *i.e.*, simultaneous arrivals and departures at the same time. We will thus be working with networks that have a constant size from the beginning to the end of a simulation.

Chapter 1

Systems with Complete Graph Structure

In this chapter, we will study open multi-agent systems with a complete structure, *i.e.*, where each agent is connected to all the others. In other words, the graph representing the system is the complete graph; see Figure 1.1. These systems are made of n agents and are considered in a discrete time. Each agent i holds at time k a value $x_i(k)$ following a uniform distribution on $[-0.5, 0.5]$. At each time step, an event occurs: either a gossip (see Definition 1.0.1) or a replacement (see Definition 1.0.2).

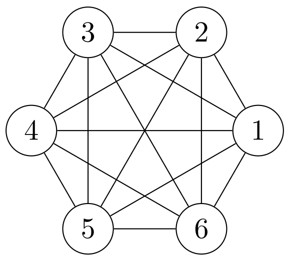


Figure 1.1: The complete graph with 6 nodes.

Definition 1.0.1. *A* ***gossip*** *is a communication between a given number of agents. When it occurs, agents implied in it will update their value following a rule defined by the system.*

Definition 1.0.2. *A* ***replacement*** *corresponds to an agent that leaves the system whilst a new one comes into the system. This new agent holds a value following the same probabilistic rule as the other agents of the system and creates connections to some agents that are already in the system. In the case of a complete graph structure, there will be an edge between this new agent and all the others. In this document, the new agent keeps always the links of the agent that has left the system. Thus, the structure does not change.*

The aim is studying the behaviour of the variance of the system following different definitions of gossips. To achieve this, we will deepen results of [HM17]. In this article, authors have worked on open multi-agent systems where gossips are defined between two agents as follows: randomly select two agents from the system i and j. Then, they will perform a pairwise average as follows:

$$x_i(k+1) = x_j(k+1) = \frac{x_i(k) + x_j(k)}{2}. \tag{1.1}$$

From this definition, we have to generate a rule for the gossips that we can use for any graph. Indeed, selecting randomly two agents in a complete graph can be seen by two different ways: In graph $\mathcal{G}$ where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges

1. Randomly select a node $i \in \mathcal{V}$. Then select randomly a node $j \in \mathcal{V}$ such that $(i, j) \in \mathcal{E}$.

2. Randomly select an edge in the graph. The two adjacent agents will be these of the gossip. In this definition an edge will be selected with a probability $\frac{1}{|\mathcal{E}|}$ where $|\mathcal{E}|$ is the cardinal of $\mathcal{E}$, *i.e.*, the number of elements of $\mathcal{E}$.

Through this work, it is the second definition that will be used. Note that this definition could be seen as a contradiction with the decentralized property. Indeed, it requires to know the number of edges, which is a very strong condition. To cope with this issue, we assume that a Poisson clock (Definition 1.0.3) is assigned to each edge. All the Poisson clocks have the same parameter. When a gossip occurs, the clock that is ringing first will define the edge selected for the gossip. Since each edge has the same Poisson clock, they all have the same probability of being selected which is equivalent to a probability $\frac{1}{|\mathcal{E}|}$.

Definition 1.0.3. *A* ***Poisson clock*** *is a clock whose rings occur following a Poisson process.*

Before going further, we will introduce a little the main results of [HM17]. One of the most useful point of this article for this work, is the notion of steady state equation (Definition 1.0.4). In [HM17], the steady state equation is provided for complete graphs when gossips are between two agents and occur with probability $1 - p$ while replacements occur with probability p.

Definition 1.0.4. *The* ***steady state equation*** *of a system in a discrete time describes the expected behaviour of the considered system from a time step to the next one in terms of some characteristics of this system (typically: its mean, mean square value, square mean value, etc).*

Proposition 1.0.1. *The steady state equation (Definition 1.0.4) of a complete graph made of n agents where a gossip between two agents defined like in (1.1) occurs with probability $1 - p$ and replacements occur with probability p is given by:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} 1 - \frac{2p}{n} & \frac{p}{n^2} \\ \frac{2(1-p)}{n} & 1 - \frac{2-p}{n} \end{pmatrix} \mathbb{E}X(k) + \sigma^2 \begin{pmatrix} \frac{p}{n^2} \\ \frac{p}{n} \end{pmatrix}, \tag{1.2}$$

where $X(k) = (\overline{x}^2(k), \overline{x^2}(k))^\top$.

Proof. The proof of this statement can be found in [HM17]. ☐

In the next sections we will first study what happens when the number of agents implied in a gossip increases. We will analyze the behaviour of the considered system when the probability associated to a replacement grows with the number of agents implied in a gossip. Then we will see how the system behaves when considering gossips with two agents where one has a more important opinion in the communication.

1.1 Gossip with Three Agents

In this section, we consider that gossips occur between three agents at a time. We will be focused on a fixed-size complete graph in which a replacement, *i.e.*, a departure and an arrival at the same time, occurs with probability $p \in [0, 1]$ and a gossip with probability $1 - p$.

To begin, we will study the behaviour of the system when replacements and gossips occur separately.

1.1.1 Gossip

We will first assume that, at each time, a gossip happens. Here, $x(k)$ is the state of the considered system at time $k \geq 0$ and $x(k+1)$ is the state at time $k+1$ after that an event occurs. In this situation the state of the three agents h, i, j exchanging their value at time k becomes:

$$x_h(k+1) = x_i(k+1) = x_j(k+1) = \frac{x_h(k) + x_i(k) + x_j(k)}{3}. \tag{1.3}$$

For notation simplicity we denote a vector

$$X(k) = (\overline{x}^2(k), \overline{x^2}(k))^\top$$

so that $\mathrm{Var}(x(k)) = (-1, 1)X(k)$.
It follows that:

Proposition 1.1.1. *The steady state equation (Definition 1.0.4) of a system made of n agents where only gossips between three agents defined like in* (1.3) *occur, is given by:*

$$\mathbb{E}X(k+1) = A_g \mathbb{E}X(k), \text{ where } A_g = \begin{pmatrix} 1 & 0 \\ \frac{2}{n} & 1 - \frac{2}{n} \end{pmatrix}, \tag{1.4}$$

and as a consequence

$$\mathbb{E}\mathrm{Var}(x(k+1)) = \left(1 - \frac{2}{n}\right) \mathbb{E}\mathrm{Var}(x(k)). \tag{1.5}$$

Proof. Given three nodes h, i and j involved in the gossip at time k. We know that $x_h(k+1) + x_i(k+1) + x_j(k+1) = \frac{x_h(k)+x_i(k)+x_j(k)}{3} \cdot 3 = x_h(k) + x_i(k) + x_j(k)$ and $x_l(k+1) = x_l(k)$, for all $l \neq h, i, j, l \in \mathcal{V}$. Hence $\overline{x}(k+1) = \overline{x}(k)$, which establishes the first line of (1.4). For the second line, since $x_l(k+1) = x_l(k)$ for every $l \neq h, i, j$, there holds

$$\begin{aligned} \overline{x^2}(k+1) &= \frac{1}{n}\sum_{l=1}^{n} x_l^2(k+1) = \overline{x^2}(k) + \frac{1}{n}\left(3\left(\frac{x_h(k) + x_i(k) + x_j(k)}{3}\right)^2 - x_h^2(k) - x_i^2(k) - x_j^2(k)\right) \\ &= \overline{x^2}(k) + \frac{2}{3n}\Big(x_h(k)x_i(k) + x_h(k)x_j(k) + x_i(k)x_j(k) - x_h^2(k) - x_i^2(k) - x_j^2(k)\Big) \end{aligned}$$

Knowing that $\mathbb{E}(x_h^2(k)|x(k)) = \mathbb{E}(x_i^2(k)|x(k)) = \mathbb{E}(x_j^2(k)|x(k)) = \overline{x^2}(k)$ and $\mathbb{E}(x_h(k)x_i(k)|x(k)) = \mathbb{E}(x_h(k)x_j(k)|x(k)) = \mathbb{E}(x_i(k)x_j(k)|x(k)) = \overline{x}^2(k)$. Taking the expectation with respect to h, i and j yields

$$\mathbb{E}(\overline{x^2}(k+1)|x(k)) = \left(1 - \frac{2}{n}\right)\overline{x^2}(k) + \frac{2}{n}\overline{x}^2$$

from which the second line of (1.4) follows. Compared to the steady state equation presented in [HM17], the first line stays the same while the second has the same structure but the $\frac{1}{n}$ of [HM17] becomes $\frac{2}{n}$. □

1.1.2 Replacement

Now, we will consider that the only event that can happen at each time step is a replacement. Since our system has a fixed size, each time an agent leaves it, it is immediately replaced by another one. In this case we speak about replacement. As previously, $x(k)$ and $x(k+1)$ respectively denote the states before and after the replacement. Then from [HM17], we know that:

$$\mathbb{E}X(k+1) = A_r \mathbb{E}X(k) + b_r, \tag{1.6}$$

where

$$A_r = \begin{pmatrix} \frac{n-2}{n} & \frac{1}{n^2} \\ 0 & \frac{n-1}{n} \end{pmatrix} \text{ and } b_r = \begin{pmatrix} \frac{\sigma^2}{n^2} \\ \frac{\sigma^2}{n} \end{pmatrix}.$$

with $\sigma^2 = \mathbb{E}x^2(k+1)$.

1.1.3 Gossip and replacement

Now that we have presented the main properties of both gossips and replacements, we can consider a graph of fixed size for which each time step involves either a replacement with probability $p \in [0,1]$ or a gossip with probability $1-p$. As in the previous sections, $x(k)$ is the state before the event and $x(k+1)$ is the state after the event.

Proposition 1.1.2. *The steady state equation (Definition 1.0.4) for a system made of n agents with probability* p *that a replacement occurs is:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} 1-\frac{2p}{n} & \frac{p}{n^2} \\ \frac{2(1-p)}{n} & 1-\frac{2-p}{n} \end{pmatrix} \mathbb{E}X(k) + \sigma^2 \begin{pmatrix} \frac{p}{n^2} \\ \frac{p}{n} \end{pmatrix}. \tag{1.7}$$

Proof. Since the events are independent of $x(k)$, the conditional expected value is computed as follows:

$$\begin{aligned} \mathbb{E}(X(k+1)|x(k)) &= (1-p)\mathbb{E}(X(k+1)|\text{gossip},x(k)) + p\mathbb{E}(X(k+1)|\text{repl},x(k)) \\ &= ((1-p)A_g + pA_r)X(k) + pb_r. \end{aligned}$$

Therefore, there holds $\mathbb{E}X(k+1) = ((1-p)A_g + pA_r)\mathbb{E}X(k) + pb_r$, which yields (1.7). □

One can verify that the fixed point (see Definition 1.1.1) of (1.7) is

$$\mathbb{E}(\overline{x})^2|_{\text{eq}} = \frac{1}{2n-1-p(n-1)}\sigma^2, \tag{1.8}$$

$$\mathbb{E}(\overline{x^2})|_{\text{eq}} = \frac{(n-1)p+1}{2n-1-p(n-1)}\sigma^2, \tag{1.9}$$

leading to a variance $\mathbb{E}\left(\text{Var}(x)\right)|_{\text{eq}} = \frac{(n-1)p}{2n-1-p(n-1)}\sigma^2$.

Definition 1.1.1. *A* ***fixed point*** *of a system such as presented in Equations* (1.2) *or* (1.7)*, is the asymptotic state of $X(k)$ when time tends to infinity,* i.e., *when $k \to +\infty$.*

1.1.4 Simulations

Equation (1.7) gives the behaviour that a system of n agents with probability of replacement p should follow. Now we can wonder how some systems behave in practice. First of all, let us define the parameters we used: the probability for a replacement is $p = 0.05$, the number of agents $n = 50$, the variance associated to the distribution of the value hold by each agent $\sigma^2 \approx \frac{1}{12}$ (see Appendix A) and the number of time steps is 10000.

From now on, in order to draw better conclusions, we will define a new measure that we can obtain from our results, namely the time-average.

Definition 1.1.2. *The* ***time-average*** *of a vector $x(k) \in \mathbb{R}^n$ at time k, $\overline{x_t}(k)$, will be defined by the following:*

$$\overline{x_t}(k) = \sum_{k'=1}^{k} \frac{x(k')}{k} \quad \forall k \in \mathbb{N}_0.$$

With this definition, we can draw the following recursive equation:

$$\overline{x_t}(k+1) = \frac{k \cdot \overline{x_t}(k) + x(k+1)}{k+1} \quad \forall k \in \mathbb{N}_0.$$

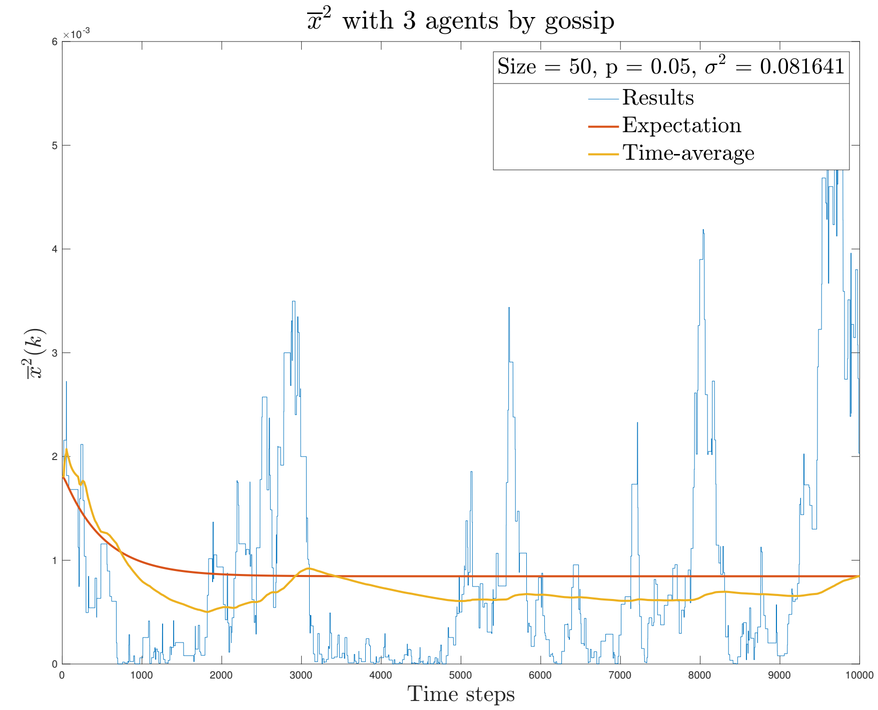


Figure 1.2: In red, the expected square mean value for gossips between three agents on 10 000 time steps using Equation (1.7). In blue the true square mean value (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

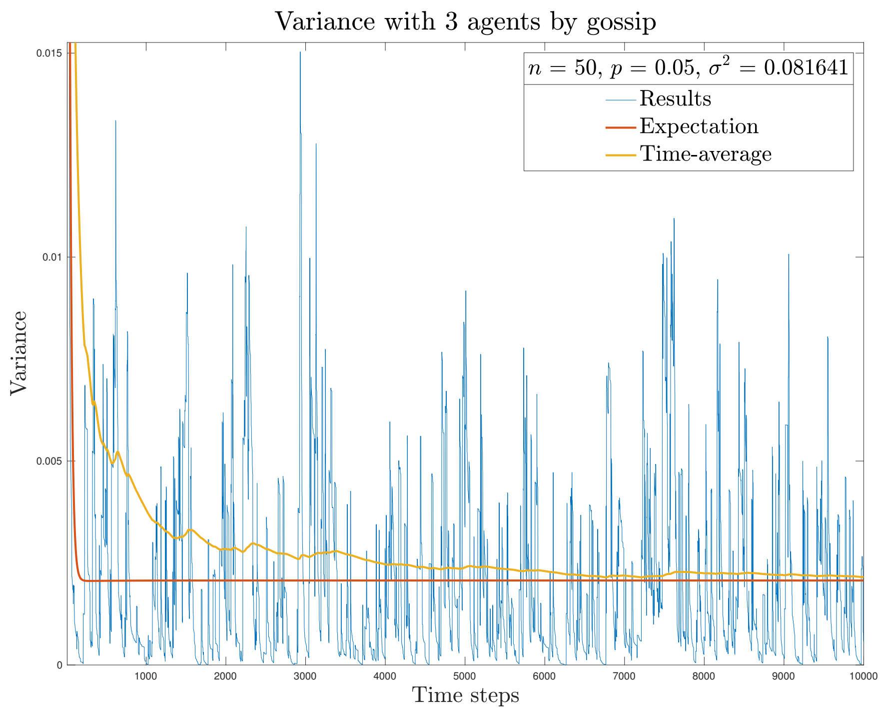


Figure 1.3: In red, the expected variance for gossips between three agents on 10 000 time steps using Equation (1.7). In blue the true variance (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve

Not only it smooths the variations from a time step to the next one, but also it describes better the behaviour of our system and results are more readable. Note that in the above definition, the time-average takes care about all the previous values of x. It is also possible to choose a smaller time-window (Definition 1.1.3), *i.e.*, to consider fewer values for the computation of the mean in order to attribute interest to only values of x that are in the time-window.

Definition 1.1.3. *The **time-window** of the time-average is the number of previous elements that comes into the computation of the mean at that time.*

For example, let us consider a window of size w, then the time-average associated will be:

$$\overline{x_t}(t+1) = \frac{\sum_{i=0}^{w-1} x(t+1-i)}{w}.$$

In Figure 1.2, we obtain the same kind of results as for a gossip with two agents, see [HM17]. Indeed, the results for our test confirm the expected behaviour quite closely. It is important to precise that the expectation reaches the value of the fixed point obtained previously when the number of time steps becomes large enough. The time-average of the variance observed for our system follows the red curve corresponding to the expected values. In Figure 1.3, we see also that our results fit in average with the expected behaviour of our system. In this context, we say that our system is ergodic. We will not study this concept in the scope of this file, readers who want to have more information about this concept are invited to consult [RFTI15].

In both cases, we observe that the time-average offers a better comprehension of the behaviour of our system. It allows us to be convinced that the variance and the square mean value reach the expected fixed point that we computed in Subsection 1.1.3.

1.2 Gossip with n_g Agents

In the previous section, we analyze the behaviour of our system when gossips were with three agents with a probability $1-p$. It is obvious that if we compare systems where gossips are with 2 or 3 agents with the same probability that a replacement or a gossip occurs, the one with 3 agents gossiping will converge faster towards a stable behaviour, *i.e.*, a state where all the agents are holding the same value.

Here we want to change the probability that a gossip occurs following the number of agents implied, and we want to compare how does the system evolve. To achieve this, we modify the way events happen. Let us now consider two Poisson processes. The first one is for replacements and has a parameter λ_r ($\mathcal{P}(\lambda_r)$) and the second one is for gossips and is characterized by λ_g ($\mathcal{P}(\lambda_g)$). We know that the sum of two independent Poisson processes is a Poisson process, characterized by $\lambda_r + \lambda_g$ ($\mathcal{P}(\lambda_r + \lambda_g)$). Hence, when an event occurs, it will be a replacement with probability $\frac{\lambda_r}{\lambda_g+\lambda_r}$ and it will be a gossip with probability $\frac{\lambda_g}{\lambda_g+\lambda_r}$.

1.2.1 Gossip

Denote $x(k)$ the state before the event, $x(k+1)$ the state after.

Proposition 1.2.1. *The steady state equation (Definition 1.0.4) for a system made of n agents where gossips are between n_g agents ($n_g \leq n$) between time k and $k+1$ ($k \geq 0$) is:*

$$\mathbb{E}X(k+1) = A_g \mathbb{E}X(k), \text{ where } A_g = \begin{pmatrix} 1 & 0 \\ \frac{n_g-1}{n} & 1-\frac{n_g-1}{n} \end{pmatrix}. \tag{1.10}$$

Proof. Given n_g nodes $i_1, \ldots, i_{n_g}$ involved in the gossip. We know that $\sum_{l=1}^{n_g} x_{i_l}(k+1) = \frac{\sum_{l=1}^{n_g} x_{i_l}(k)}{n_g} n_g = \sum_{l=1}^{n_g} x_{i_l}(k)$ and $x_m(k+1) = x_m(k)$, for all $m \neq i_l \quad \forall l = 1, \ldots, n_g$ and $m \in \mathcal{V}$. Hence $\overline{x}(k+1) = \overline{x}(k)$, which establishes the first line of (1.10). For the second line, since $x_m(k+1) = x_m(k)$ for every $m \neq i_l \quad \forall l = 1, \ldots, n_g$ and $m \in \mathcal{V}$, there holds

$$\begin{aligned}\overline{x^2}(k+1) &= \frac{1}{n}\sum_{i=1}^{n} x_i^2(k+1) = \overline{x^2}(k) + \frac{1}{n}\left(n_g\left(\frac{\sum_{l=1}^{n_g} x_{i_l}(k)}{n_g}\right)^2 - \sum_{l=1}^{n_g} x_{i_l}^2(k)\right) \\ &= \overline{x^2}(k) + \frac{1}{n}\left(\frac{1}{n_g}\sum_{l=1}^{n_g} x_{i_l}(k) \sum_{l'=1, l'\neq l}^{n} x_{i_{l'}}(k) - \frac{n_g-1}{n_g}\sum_{l=1}^{n_g} x_{i_l}^2(k)\right).\end{aligned}$$

Knowing that $\mathbb{E}(x_{i_l}^2(k)|x(k)) = \overline{x^2}(k), \forall l = 1, \ldots, n_g$ and $\mathbb{E}(x_{i_l}(k)x_{i_{l'}}(k)|x(k)) = \overline{x}^2(k), \forall l, l' = 1, \ldots, n_g$ and $l \neq l'$. Taking the expectation with respect to i_1, i_2, $\ldots$, i_{n_g} yields

$$\mathbb{E}(\overline{x^2}(k+1)|x(k)) = \left(1 - \frac{n_g-1}{n}\right)\overline{x^2}(k) + \frac{n_g-1}{n}\overline{x}^2(k)$$

from which the second line of (1.10) follows. $\square$

1.2.2 Replacement

The steady state equation for this part is exactly the same as the one in Subsection 1.1.2. Indeed, if we modify the way gossips occur, we do not change how a replacement is defined.

1.2.3 Gossip and replacement

In the previous sections, a system was defined with a number of agents and a given probability that a replacement occurs. Now, we want to compare the behaviour of a system when gossips imply n_{g_1} agents with probability

$$\frac{\lambda_{n_{g_1}}}{\lambda_{n_{g_1}} + \lambda_r},$$

and n_{g_2} agents with probability

$$\frac{\lambda_{n_{g_2}}}{\lambda_{n_{g_2}} + \lambda_r}.$$

We want to investigate about the best number of agents implied in a gossip. Comparing results obtained with the system where n_{g_1} and n_{g_2} agents are in a gossip allows us to study how the variance and the square mean value of the considered system change. The best choice will be the one that gives the smallest fixed point and that will require the least possible computations. Since we want that the same number of agents has made a gossip on a given interval of time, we add the following constraint:

$$n_{g_1}\lambda_{n_{g_1}} = n_{g_2}\lambda_{n_{g_2}}. \tag{1.11}$$

Then we have two different steady state equations following that n_{g_1} or n_{g_2} agents are implied in a gossip. If we want to study the behaviour of the considered system, we have to introduce the reference value for $n_{g_1}\lambda_{n_{g_1}}$. If the number of agents in a gossip changes, then the probability that a gossip occurs changes following the reference value and Equation (1.11).

Proposition 1.2.2. *The steady state equation (Definition 1.0.4) for a system where a gossip implies n_g agents and occurs with a probability $\frac{\lambda_g}{\lambda_g+\lambda_r}$ and a replacement occurs with a probability $\frac{\lambda_r}{\lambda_g+\lambda_r}$ is given by:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} 1 - \frac{2\lambda_r}{n(\lambda_g+\lambda_r)} & \frac{\lambda_r}{(\lambda_g+\lambda_r)n^2} \\ \frac{(n_g-1)\lambda_g}{(\lambda_g+\lambda_r)n} & 1 - \frac{(n_g-1)\lambda_g+\lambda_r}{(\lambda_g+\lambda_r)n} \end{pmatrix} \mathbb{E}X(k) + \sigma^2 \begin{pmatrix} \frac{\lambda_r}{(\lambda_g+\lambda_r)n^2} \\ \frac{\lambda_r}{(\lambda_g+\lambda_r)n} \end{pmatrix}. \tag{1.12}$$

Proof. Since the events are independent of $x(k)$, the conditional expected value is computed as follows:

$$\begin{aligned}\mathbb{E}(X(k+1)|x(k)) &= \frac{\lambda_g}{\lambda_g+\lambda_r}\mathbb{E}(X(k+1)|\text{gossip},x(k)) + \frac{\lambda_r}{\lambda_g+\lambda_r}\mathbb{E}(X(k+1)|\text{repl},x(k)) \\ &= \left(\frac{\lambda_g}{\lambda_g+\lambda_r}A_g + \frac{\lambda_r}{\lambda_g+\lambda_r}A_r\right)\mathbb{E}X(k) + \frac{\lambda_r}{\lambda_g+\lambda_r}b_r.\end{aligned}$$

Therefore, there holds $\mathbb{E}X(k+1) = \left(\frac{\lambda_g}{\lambda_g+\lambda_r}A_g + \frac{\lambda_r}{\lambda_g+\lambda_r}A_r\right)\mathbb{E}X(k) + \frac{\lambda_r}{\lambda_g+\lambda_r}b_r$, which yields (1.12). □

1.2.4 Fixed point

Equation (1.12) gives an idea of how the system changes at each event. It is nevertheless legitimate to ask towards which value the system will tend asymptotically. To answer to this question we have to solve the linear system of (1.12) saying that $X(k+1) = X(k)$.

To achieve this, `Mathematica` has been used.

Proposition 1.2.3. *The fixed point (Definition 1.1.1) of the steady state equation* (1.12) *is given by:*

$$\mathbb{E}(\overline{x})^2|_{eq} = \frac{(2\lambda_r + (n_g-1)\lambda_g)\sigma^2}{2\lambda_r n + \lambda_g(1-n_g)(1-2n)}, \tag{1.13}$$

$$\mathbb{E}(\overline{x^2})|_{eq} = \frac{(2\lambda_r n + (n_g-1)\lambda_g)\sigma^2}{2\lambda_r n + \lambda_g(1-n_g)(1-2n)}. \tag{1.14}$$

A study of the value of (1.13) and (1.14) is quite interesting. In order to see how it changes, we will fix a Poisson parameter for the replacement, say λ_r, a Poisson parameter for the gossip, λ_g. In Figures 1.4 and 1.5 we observe the fixed point of the square mean value and the variance respectively as a function of the number of agents that are engaged in a gossip with the constraint introduced in Equation (1.11).

In our case, we have chosen $\lambda_r = 5$, $\lambda_g(t=0) = 95$, $n_g(t=0) = 2$ and $n_g(t+1) = n_g(t)+1$. Initially, there is therefore probability of 5% that a replacement occurs. Obviously, increasing the number of agents implied in a gossip will increase this probability. We will use Equation (1.11) with the following parameter: $n_{g_1} = n_g(t)$, $n_{g_2} = n_g(t+1)$, $\lambda_{n_{g_1}} = \lambda_{n_g}(t)$ and $\lambda_{n_{g_2}} = \frac{\lambda_{n_{g_1}} n_{g_1}}{n_{g_2}}$. As we can see, the variance and the square mean value decrease when n_g increases. We can observe that after a certain value, both graphs have an asymptotic behaviour. It means that increasing the number of agents implied in a gossip will not decrease the fixed point of the variance and the fixed point of the square mean value of our system since increasing n_g reduces the value of λ_g. As a consequence, when an event occurs, the probability that this event is a replacement is more important. It also means that the variance and the square mean value that we will observe will be more noisy, as we can see in Figures 1.10 and 1.11. We will explain this behaviour in Subsection 1.2.5.

To have more intuition about the evolution of the fixed point, we will go further by computing its value when $n_g \to n$. In order to simplify our notation, let us define a constant $C = \lambda_g(0)n_g(0)$. Of course, by definition of constraint (1.11), C will not change with time. We thus obtain the

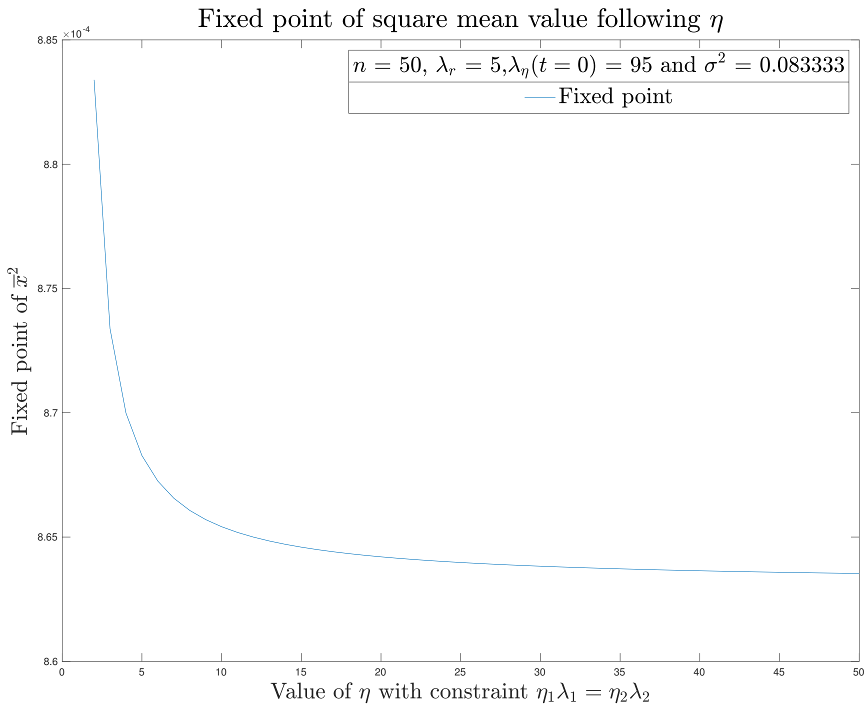


Figure 1.4: Fixed point of square mean value as a function of the number of agents implied in a gossip using Equation (1.12).

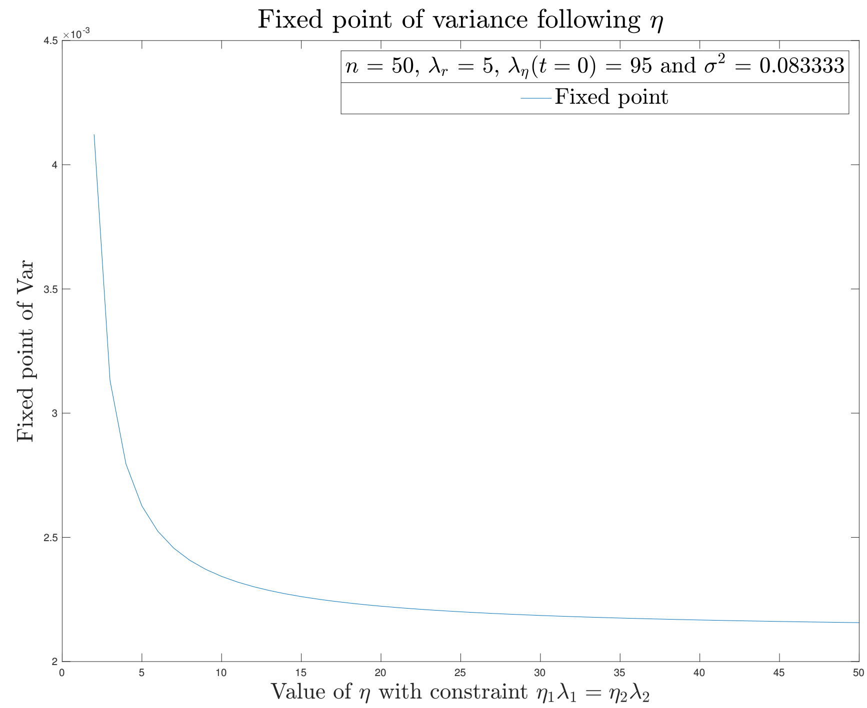


Figure 1.5: Fixed point of variance as a function of the number of agents implied in a gossip using Equation (1.12).

following results:

$$\begin{aligned}
\mathbb{E}\left(\operatorname{Var}(x)_{n_g \to n}\right)\big|_{\text{eq}} &= \left.\frac{2\lambda_r(n-1)\sigma^2}{2\lambda_r n + \lambda_g(n_g-1)(2n-1)}\right|_{n_g \to n} \\
&= \frac{2\lambda_r(n-1)\sigma^2}{2\lambda_r n + C(1-\frac{1}{n})(2n-1)} \\
&\simeq \frac{2\lambda_r(n-1)\sigma^2}{2\lambda_r n + C(2n-3)}, \\
\mathbb{E}\left(\overline{x}^2_{n_g \to n}\right)\Big|_{\text{eq}} &= \left.\frac{(2\lambda_r + (n_g-1)\lambda_g)\sigma^2}{2\lambda_r n + \lambda_{n_g}(n_g-1)(2n-1)}\right|_{n_g \to n} \\
&\simeq \frac{(2\lambda_r + C)\sigma^2}{2\lambda_r n + C(2n-3)}.
\end{aligned}$$

Note that our approximations are valid only for n sufficiently large. Those results substantiate the behaviour described in Figures 1.4 and 1.5. However, a larger number of agents in our system would provide better result. Nevertheless, our solutions are already quite precise. We obtain $8.804 \cdot 10^{-4}$ for the square mean value and $2.1571 \cdot 10^{-3}$ for the variance whilst the results obtained are $8.6354 \cdot 10^{-4}$ and $2.1570 \cdot 10^{-3}$ respectively. Our error is thus less than two percent for a number of agents in our system which is quite small.

1.2.5 Simulations

In this section, we will illustrate the results obtained previously with a system composed of 50 agents where the values of agents are uniformly distributed on $[\frac{-1}{2}, \frac{1}{2}]$ and thus variance $\sigma^2 \approx \frac{1}{12}$ (see Appendix A).

In Figures 1.6 and 1.7 we show the square mean value of our system when a gossip implied $n_{g_1} = 2$ and $n_{g_2} = 10$ agents respectively. For this, we have chosen $\lambda_{n_{g_1}} = 95$, $\lambda_r = 5$ and then $\lambda_{n_{g_2}} = 19$. As expected, the more n_{g_2} is bigger than n_{g_1}, the more the square mean value moves from values that are closed to 0, to values that are significantly larger than the expected ones. It can be explained by the following: each time a gossip occurs, the mean becomes closer to 0 since the value associated to an agent is uniformly distributed with a zero mean. However, since the probability for a gossip to occur is smaller, the expected number of replacements between two gossips is greater, which explains the peaks on the graph.

Figures 1.8 and 1.9 show the variance for gossips implying $n_{g_1} = 2$ and $n_{g_2} = 10$ agents respectively. To comment those graphs, we can use the same reasoning as for the square mean value: the variance decreases easily close to 0 (gossip effect), but increases very fast since replacements are more frequent between two gossips when n_g is large compared to n.

Now to illustrate with an extreme simulation, Figures 1.10 and 1.11 show the square mean value and the variance of our system where a gossip implies 50 agents. It is the most extreme situation since here $n_g = n = 50$. As we can see, each time a gossip occurs, the variance is equal to zero and the square mean value tends to zero. Both illustrations confirm the behaviour observed previously with 10 agents in a gossip. Note that if the number of agents was sufficiently large, the square mean value after a gossip should also be equal to zero by definition of the distribution that the values held by the agents follow.

To conclude, we can say that an important number of agents implied in a gossip are favourable in order to obtain small fixed point. However, as we can see in Figures 1.4 and 1.5, the fixed point of the variance and the fixed point of the square mean value reach quickly an asymptotic behaviour. Thus, an interesting choice for the number of agents in a gossip would be selecting one such that the fixed point is quite small while the square mean value and the variance are not

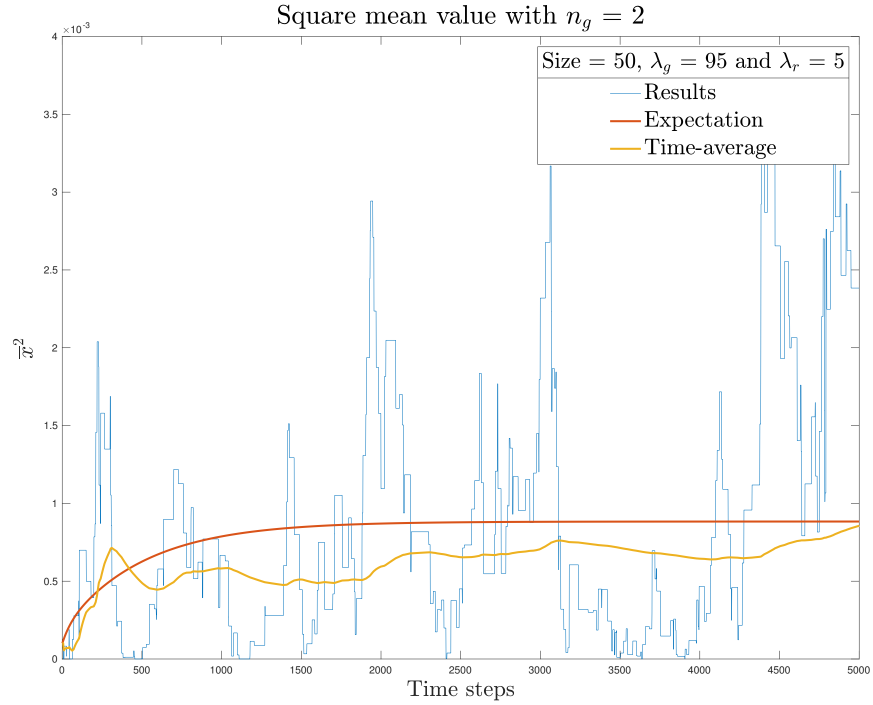


Figure 1.6: In red, the expected square mean value for gossips between 2 agents on 5000 time steps using Equation (1.12). In blue, the true square mean value (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

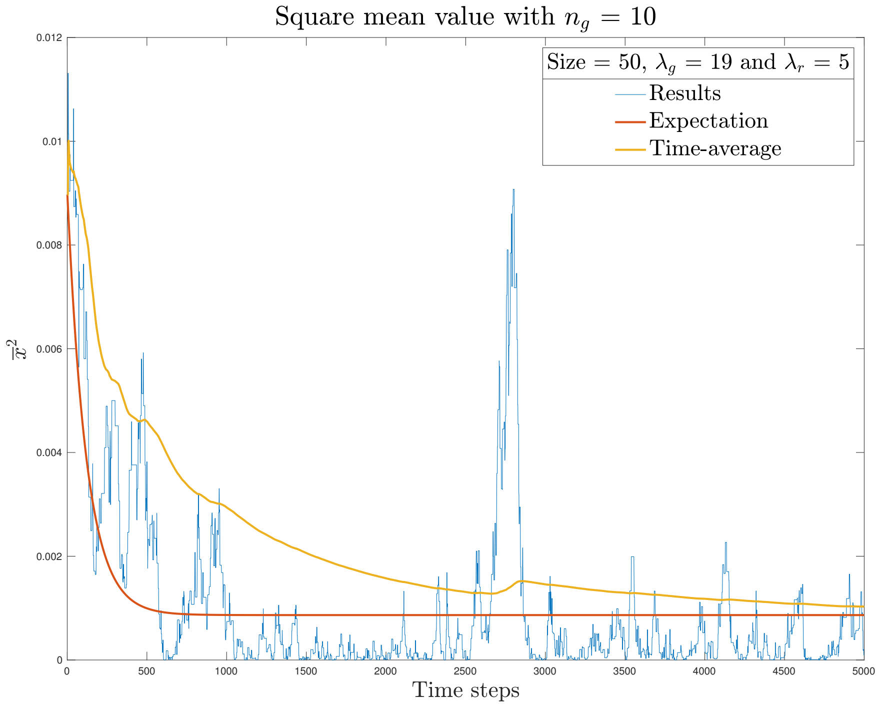


Figure 1.7: In red, the expected square mean value for gossips between 10 agents on 5000 time steps using Equation (1.12). In blue, the true square mean value (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

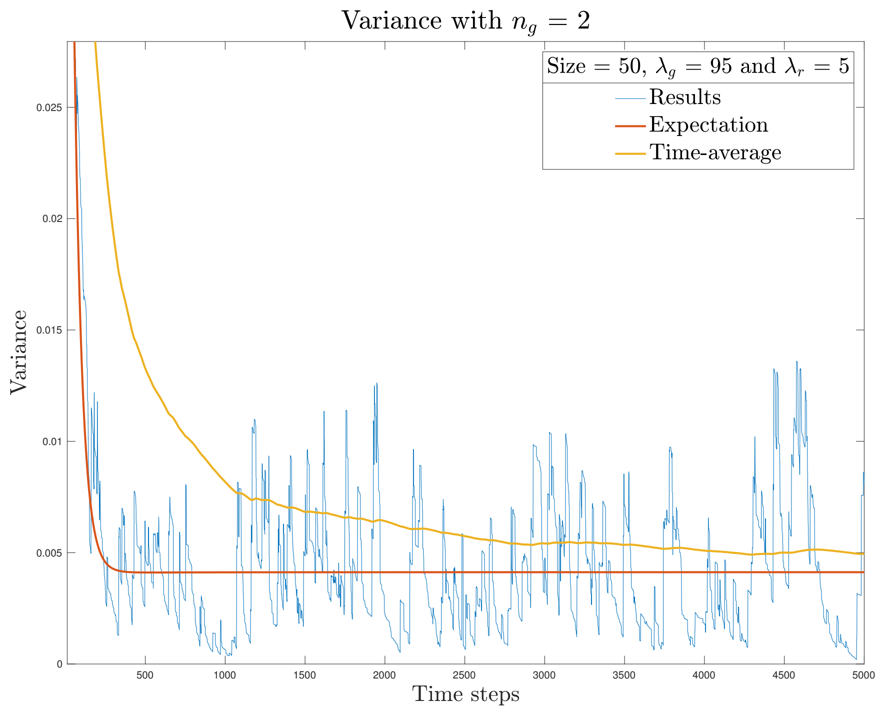


Figure 1.8: In red, the expected variance for gossips between 2 agents on 5000 time steps using Equation (1.12). In blue, the true variance (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

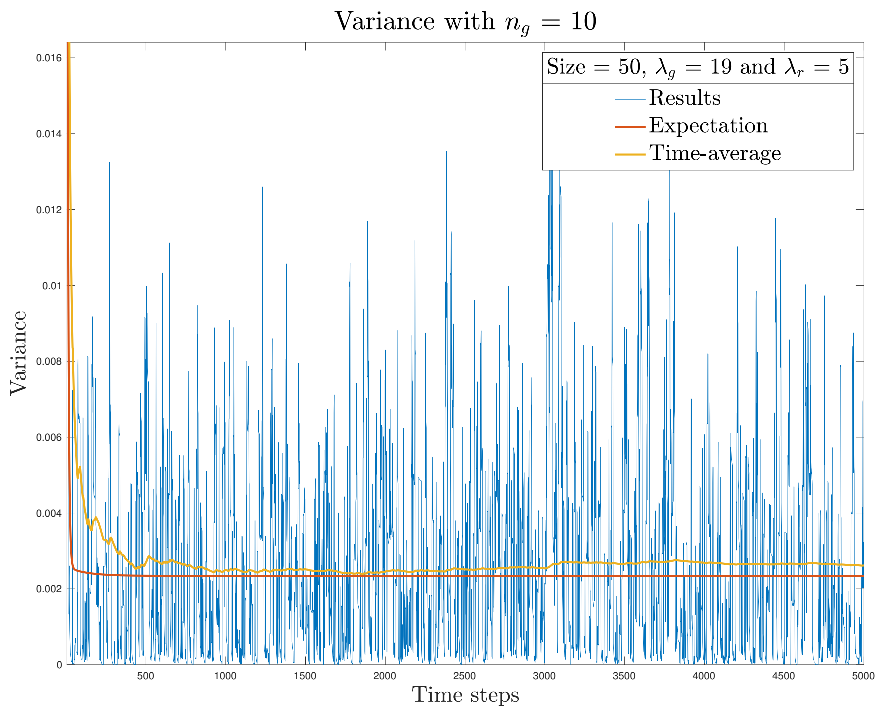


Figure 1.9: In red, the expected variance for gossips between 10 agents on 5000 time steps using Equation (1.12). In blue, the true variance (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

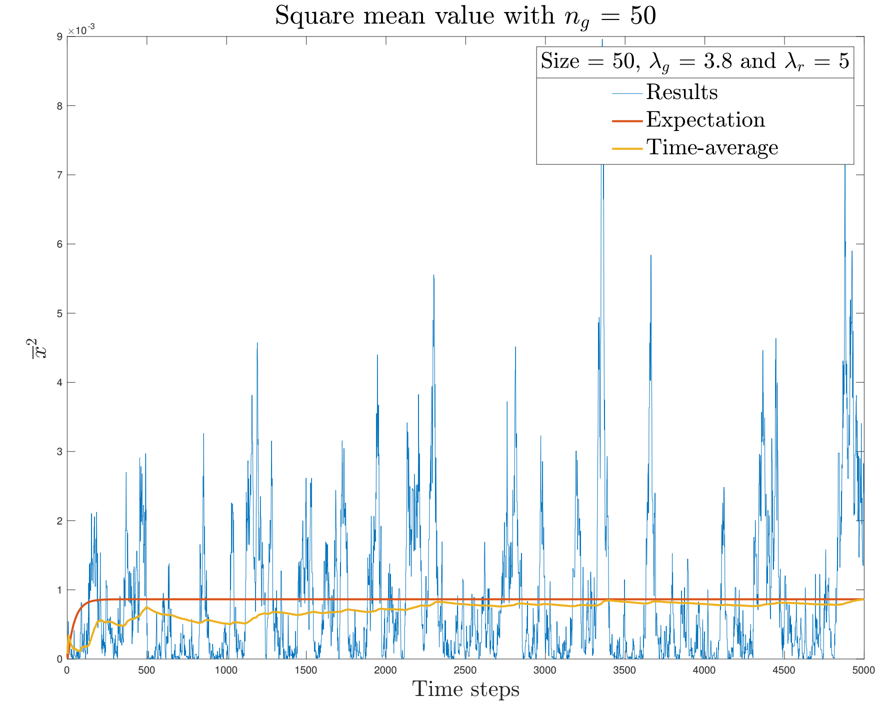


Figure 1.10: In red, the expected square mean value for gossips between 50 agents on 5000 time steps using Equation (1.12). In blue, the true square mean value (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

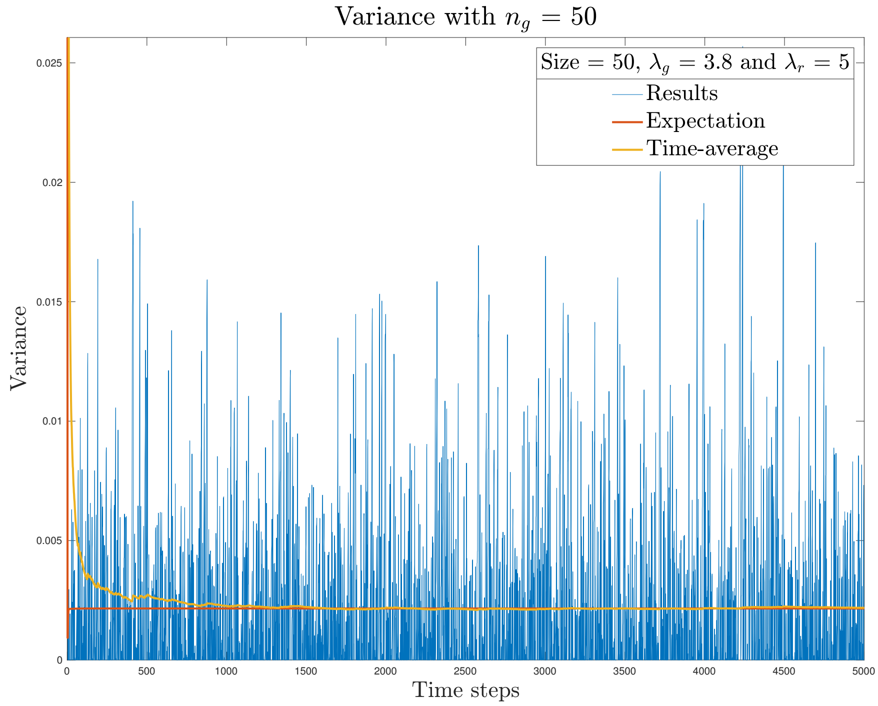


Figure 1.11: In red, the expected variance for gossips between 50 agents on 5000 time steps using Equation (1.12). In blue, the true square mean value (see Notation) of the system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

oscillating too much. In the case of our system made of 50 agents, $n_g = 10$ would for example be a reasonable choice. Another point that has not been discussed through this part and Section 1.1 is the decentralized property. When gossips are between two agents, we can say that we randomly select an edge with a probability $\frac{1}{|\mathcal{E}|}$. When gossips are between more agents, this method does not hold anymore. To cope with this, we can imagine that the agents have also a Poisson clock (Definition 1.0.3) with always the same parameter. Since we know that we are in a complete topology, each agent is connected to all the others. Thus, it is possible to choose the agents implied in gossip by selecting the n_g first agents whose Poisson clock rings.

1.3 Gossip with α-mean

Until now, each agent implied in a gossip had the same importance, *i.e.*, had the same weight in the gossip. In this section, we want to quantify how the system would evolve if in a gossip between two agents, one of them had a more important opinion. In this case, which could arise for instance in opinion dynamic models, we speak about alpha-mean (Definition 1.3.1).

Definition 1.3.1. *An **alpha-mean** update in a gossip between two agents is a way of updating their value where each agent will influence its new value with a weight α while the second one will have a weight $(1-\alpha)$. If the gossip is between agent i and j at time k, it will be defined as follows:*

$$x_i(k+1) = \alpha x_i(k) + (1-\alpha)x_j(k), \tag{1.15}$$
$$x_j(k+1) = \alpha x_j(k) + (1-\alpha)x_i(k). \tag{1.16}$$

Proposition 1.3.1. *The steady state equation (Definition 1.0.4) for a system where gossips are between two agents and following an alpha-mean is given by:*

$$\mathbb{E}X(k+1) = A_g \mathbb{E}X(k), \text{ where } A_g = \begin{pmatrix} 1 & 0 \\ \frac{4\alpha(1-\alpha)}{n} & \frac{4\alpha^2-4\alpha+n}{n} \end{pmatrix}, \tag{1.17}$$

and as a consequence

$$\mathbb{E}\mathrm{Var}(x(k+1)) = \frac{4\alpha^2-4\alpha+n}{n}\mathbb{E}\mathrm{Var}(x(k)). \tag{1.18}$$

Proof. If i and j are the two nodes implied in the gossip, then we have:

$$\overline{x}(k+1) = \overline{x}(k) + \alpha x_i(k) + (1-\alpha)x_j(k) + \alpha x_j(k) + (1-\alpha)x_i(k) = \overline{x}(k).$$

Then we have the first line of (1.17). For the second line,

$$\begin{aligned}
\overline{x^2}(k+1) &= \overline{x^2}(k) + \frac{1}{n}\Big((\alpha x_i(k) + (1-\alpha)x_j(k))^2 + (\alpha x_j(k) + (1-\alpha)x_i(k))^2 - x_i^2(k) - x_j^2(k)\Big) \\
&= \overline{x^2}(k) + \frac{1}{n}\Big(\Big(x_i^2(k) + x_j^2(k)\Big)\Big((1-\alpha)^2 + \alpha^2 - 1\Big) + 4\alpha(1-\alpha)x_i(k)x_j(k)\Big) \\
&= \overline{x^2}(k) + \frac{1}{n}\Big(\Big(x_i^2(k) + x_j^2(k)\Big)\Big(2\alpha^2 - 2\alpha\Big) + 4\alpha(1-\alpha)x_i(k)x_j(k)\Big).
\end{aligned}$$

Knowing that $\mathbb{E}(x_i^2(k)|x(k)) = \mathbb{E}(x_j^2(k)|x(k)) = \overline{x^2}(k)$ and $\mathbb{E}(x_i(k)x_j(k)|x(k)) = \overline{x}^2(k)$. Taking the expectation with respect to i, j yields

$$\mathbb{E}(\overline{x^2}(k+1)|x(k)) = \frac{1}{n}\Big((4\alpha^2 - 4\alpha + n)\overline{x^2}(k) + 4\alpha(1-\alpha)\overline{x}^2(k)\Big)$$

from which the second line of (1.17) follows. $\square$

Table 1.1: Time of convergence following α.

$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$	$\alpha = 0.7$	$\alpha = 0.8$	$\alpha = 0.9$
8770.6	5923.5	4476.0	4024.9	3742.0	4046.5	4759.4	5350.8	10 697.0

Equation (1.17) gives the same result as [HM17] when $\alpha = \frac{1}{2}$.

Proposition 1.3.2. *The steady state equation (Definition 1.0.4) assuming that the system at each step makes either a gossip following an alpha-mean (Definition 1.3.1) with probability* $(1-p)$ *or a replacement with probability* p *is:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} 1 - \frac{2p}{n} & \frac{p}{n^2} \\ \frac{4\alpha(1-\alpha)(1-p)}{n} & \frac{(n-p)+(1-p)(4\alpha^2-4\alpha)}{n} \end{pmatrix} \mathbb{E}X(k) + \sigma^2 \begin{pmatrix} \frac{p}{n^2} \\ \frac{p}{n} \end{pmatrix}. \tag{1.19}$$

Proof. It suffices to sum two matrices: A_g that we multiply by $(1-p)$ and A_r of Subsection 1.1.2 multiplied by p. $\square$

Note that when $\alpha = \frac{1}{2}$, we have the same results as in [HM17].

Now we want to find the fixed point of Equation (1.19), this is to say that we have to solve the system

$$\mathbb{E}(\overline{x})^2 = \frac{\mathbb{E}(\overline{x^2}) + \sigma^2}{2n}$$
$$\mathbb{E}(\overline{x^2})\left(p + 4\alpha(\alpha-1)(p-1)\right) = 4\alpha(1-\alpha)(1-p)\mathbb{E}(\overline{x})^2 + p\sigma^2$$

which gives:

$$\mathbb{E}(\overline{x})^2|_{\mathrm{eq}} = \frac{p + 2\alpha(\alpha-1)(p-1)}{pn + 2(\alpha-1)(p-1)(2n-1)\alpha}\sigma^2, \tag{1.20}$$

$$\mathbb{E}(\overline{x^2})|_{\mathrm{eq}} = \frac{2\alpha(1-\alpha)(1-p) + pn}{pn + 2(\alpha-1)(p-1)(2n-1)\alpha}\sigma^2. \tag{1.21}$$

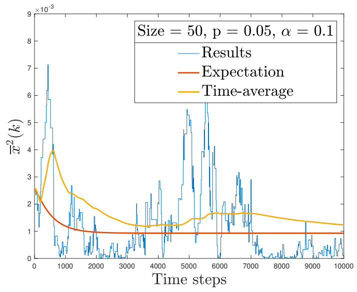
It is important to mention that if $\alpha = 0.5$, the results reduce to those of [HM17].

Figures 1.12 and 1.13 show the square mean value and the variance as a function of α respectively for a system made of 50 agents and probability for a replacement $p = 0.05$ over 10000 time steps.

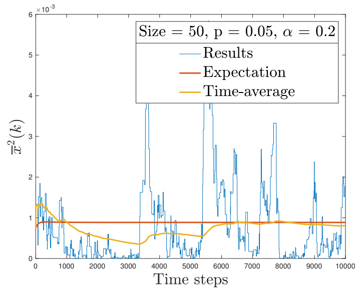
Note that the results for the variance seem to be symmetric around $\alpha = 0.5$. It does not surprise us since the matrix of Equation (1.19) was an even function around $\alpha = 0.5$.

Whatever the value of α, the mean of the system stays the same. The main difference is the time required in order to converge towards the expected behaviour. Table 1.1 shows the average time (on 10 simulations for each value of α) required such that the time-average of the variance of our system becomes equal to the expected behaviour within 10^{-3}. We can still observe the symmetry around $\alpha = 0.5$ since the time required on each side of 0.5 grows in a comparable way.

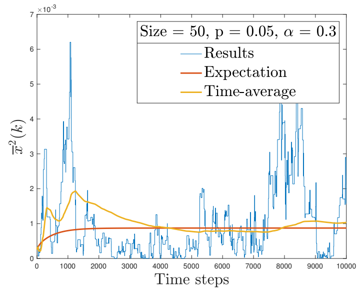
Finally, we can say that using alpha-mean for our gossip is not efficient in terms of time required for obtaining a state closed to the expected one. By definition of steady state Equation (1.19), we can observe a symmetry around the value $\alpha = 0.5$. These results are quite intuitive since we want that the value held by each agent reaches the average value of the system. In this context, it is obvious that the farther form 0.5 $\alpha \in [0, 1]$ will be, the slower our results behave as the steady state equation suggests.



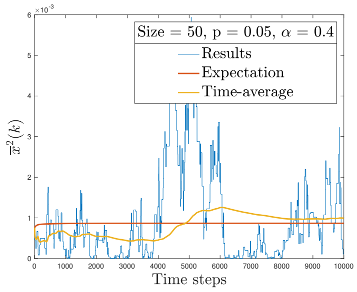
(a) $\alpha = 0.1$



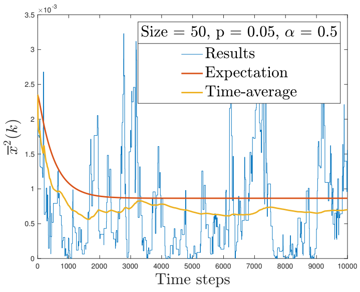
(b) $\alpha = 0.2$



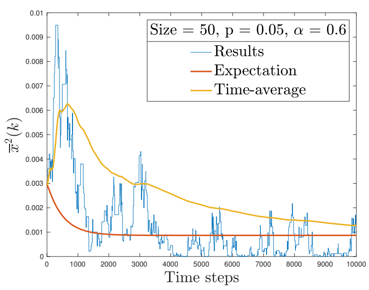
(c) $\alpha = 0.3$



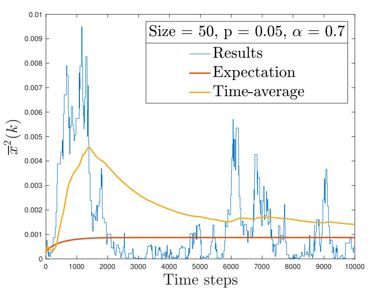
(d) $\alpha = 0.4$



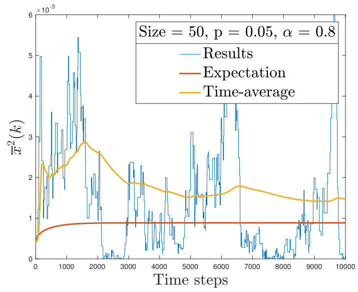
(e) $\alpha = 0.5$



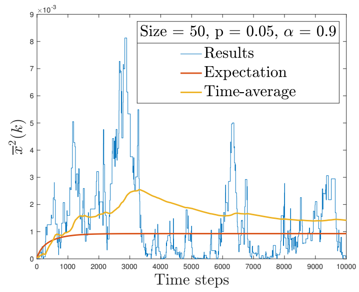
(f) $\alpha = 0.6$



(g) $\alpha = 0.7$

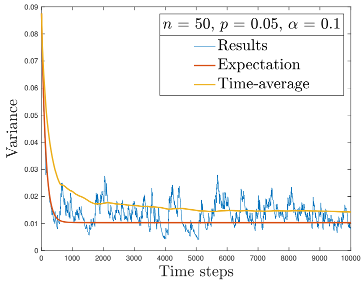


(h) $\alpha = 0.8$

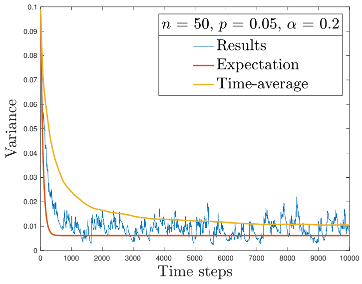


(i) $\alpha = 0.9$

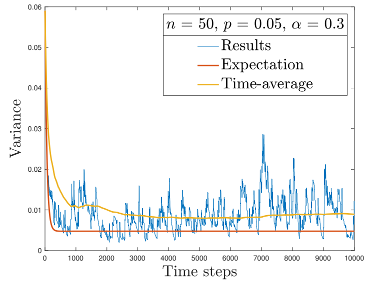
Figure 1.12: In red, the expected square mean value as a function of α on 10000 time steps using Equation (1.19). In blue, the true square mean value (see Notation) of the system at each time step following the value of α. In yellow, the time-average (Definition 1.1.2) of the blue curve.



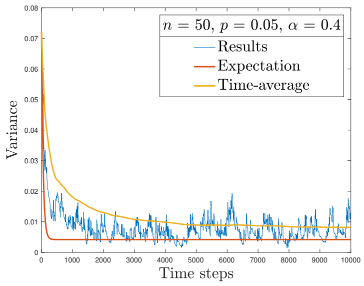
(a) $\alpha = 0.1$



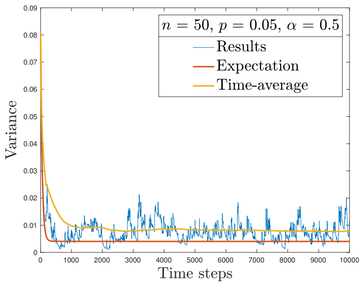
(b) $\alpha = 0.2$



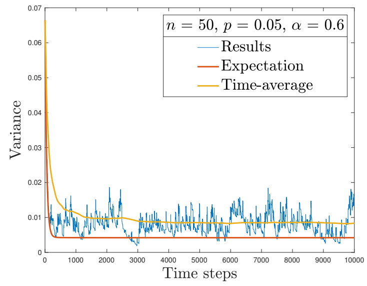
(c) $\alpha = 0.3$



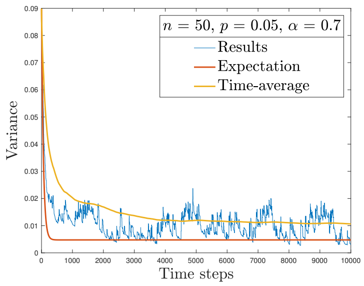
(d) $\alpha = 0.4$



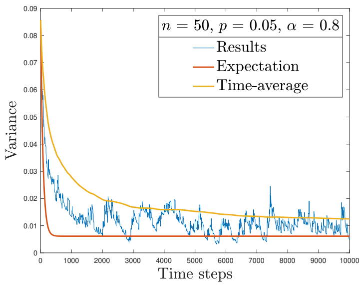
(e) $\alpha = 0.5$



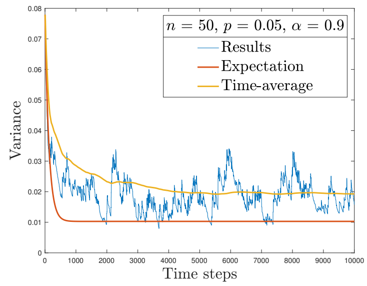
(f) $\alpha = 0.6$



(g) $\alpha = 0.7$



(h) $\alpha = 0.8$



(i) $\alpha = 0.9$

Figure 1.13: In red, the expected variance as a function of α on 10000 time steps using Equation (1.19). In blue, the true variance (see Notation) of the system at each time step following the value of α. In yellow, the time-average (Definition 1.1.2) of the blue curve.

1.4 Conclusion

In this first chapter we extended the research conducted in [HM17]. We were then working on systems that have a complete graph structure. First of all we computed the steady state equation when a gossip is between three agents. Obviously, in this configuration the considered system reaches faster its fixed point than when gossips are between only two agents.

The second section aimed to study systems where the probability that a gossip occurs depends on the number of agents that are implied in the system. The goal was to find the most efficient gossip such that the considered system reaches quickly the expected behaviour. From those researches, we have concluded that taking a number too large is not the best idea since the square mean value and the variance oscillate too much while taking a value too small is not the best choice if we want to obtain a fixed point as small as possible.

Finally, in the third section, we saw that when a gossip occurs, the best way to update the value of the two agents that are implied is by taking the arithmetic mean of their value. Indeed, selecting another weight distribution will slow down the convergence towards the fixed point of the system.

In the next chapters, we will try to be more general, and we will be working on graphs that are not complete anymore. As we did in this chapter, we will compute the steady state equation of some particular graphs, and we will test the obtained results.

Chapter 2

Systems with Complete Bipartite Graph Structure

In this chapter, we will try to reproduce the same results as in Chapter 1, but with a particular kind of graphs that are not complete, namely complete bipartite graphs. We will consider gossips as they are defined in Equation (1.1). The aim of the three first sections is obtaining the steady state equation (Definition 1.0.4) of such graphs. From now on, the vector $X(k)$ will no longer depend on the whole system but will consider each partition $\mathcal{V}_a$ and $\mathcal{V}_b$ of the graph separately.

Then we will be looking for the fixed point of this kind of graphs and a comparison between the results for the complete bipartite graph and the complete graph will be done. We will also quickly study how to link the behaviour of a partition of the graph to the whole system. In order to illustrate our researches, simulations will be done for a better understanding of the evolution of such a sort of systems.

Finally, we will be focused on a particular case of bipartite graph, namely the star graph. The steady state equation and the fixed point for this sort of bipartite graph will be provided. The last section will be dedicated to the following question: *which agent to look at if we want to have the best approximation of the behaviour of our graph?*

2.1 Complete Bipartite Graph

Definition 2.1.1. *A **bipartite graph** is a simple graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ in which $\mathcal{V}$ can be partitioned into two sets, $\mathcal{V}_a$ and $\mathcal{V}_b$ with the following properties:*

1. *if $v \in \mathcal{V}_a$ then it may only be adjacent to vertices in $\mathcal{V}_b$;*
2. *if $v \in \mathcal{V}_b$ then it may only be adjacent to vertices in $\mathcal{V}_a$;*
3. *$\mathcal{V}_a \cap \mathcal{V}_b = \emptyset$;*
4. *$\mathcal{V}_a \cup \mathcal{V}_b = \mathcal{V}$.*

[Sal07]

A complete bipartite graph is a bipartite graph where each node of a partition is connected to all the nodes of the other partition. An example of this sort of graphs has been drawn in Figure 2.1.

First of all we want to find the steady state equation of such graphs. To achieve this, $X(k)$ will now take into account each partition of the graph such that:

$$X(k) = \begin{pmatrix} \overline{x_a}^2(k) & \overline{x_b}^2(k) & \overline{x_a^2}(k) & \overline{x_b^2}(k) & \overline{x_a}(k)\overline{x_b}(k) \end{pmatrix}^\top,$$

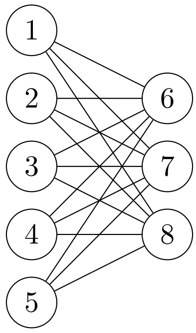


Figure 2.1: A complete bipartite graph.

where $\overline{x_a}(k)$ is the mean of $\mathcal{V}_a$ at time k, $\overline{x_b}(k)$ is the mean of $\mathcal{V}_b$ at time k, $\overline{x_a^2}(k)$ is the mean square value of $\mathcal{V}_a$ at time k and $\overline{x_b^2}(k)$ is the mean square value of $\mathcal{V}_b$ at time k.

We also assume that we have a complete bipartite graph with partitions $\mathcal{V}_a$ and $\mathcal{V}_b$. In this context, a gossip will imply two agents that come from different partitions and are defined as in Equation (1.1). $x(k)$ is the state of the system at time k and $x(k+1)$ is its state at time $k+1$. With the same reasoning, $x_a(k)$ and $x_b(k)$ are the states of $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively, before the event while $x_a(k+1)$ and $x_b(k+1)$ are the states of $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively, after the event. n_a and n_b are the numbers of agents in partitions $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively.

2.1.1 Gossip

Proposition 2.1.1. *The steady state equation (Definition 1.0.4) of a complete bipartite graph when only gossips from* (1.1) *occur at time k and where $x(k)$ is the state of the system at time k is given by:*

$$\mathbb{E}X(k+1) = A_g \mathbb{E}X(k),\ \textit{where}\ A_g = \begin{pmatrix} 1-\frac{1}{n_a} & 0 & \frac{1}{4n_a^2} & \frac{1}{4n_a^2} & \frac{2n_a-1}{2n_a^2} \\ 0 & 1-\frac{1}{n_b} & \frac{1}{4n_b^2} & \frac{1}{4n_b^2} & \frac{2n_b-1}{2n_b^2} \\ 0 & 0 & 1-\frac{3}{4n_a} & \frac{1}{4n_a} & \frac{1}{2n_a} \\ 0 & 0 & \frac{1}{4n_b} & 1-\frac{3}{4n_b} & \frac{1}{2n_b} \\ \frac{1}{2n_b} & \frac{1}{2n_a} & \frac{-1}{4n_a n_b} & \frac{-1}{4n_a n_b} & 1-\frac{n_a+n_b-1}{2n_a n_b} \end{pmatrix}. \tag{2.1}$$

Proof. Let us first fix the node $i_a \in \mathcal{V}_a$ and $i_b \in \mathcal{V}_b$ involved in the gossip. Observe that $x_l(k+1) = x_l(k)\ \forall l \neq i_a,\ l \in \mathcal{V}_a$ and $x_m(k+1) = x_m(k)\ \forall m \neq i_b,\ m \in \mathcal{V}_b$. Then we have:

$$\overline{x_a}(k+1) = \frac{1}{n_a}\sum_{l=1}^{n_a} x_l(k+1) = \overline{x_a}(k) + \frac{1}{2n_a}\left(x_{i_b}(k) - x_{i_a}(k)\right),$$

thus,

$$\overline{x_a}^2(k+1) = \overline{x_a}^2(k) + \frac{\overline{x_a}(k)}{n_a}\left(x_{i_b}(k) - x_{i_a}(k)\right) + \frac{1}{4n_a^2}\left(x_{i_a}^2(k) + x_{i_b}^2(k) - 2x_{i_a}(k)x_{i_b}(k)\right),$$

and taking the expectation we obtain:

$$\mathbb{E}\left(\overline{x_a}^2(k+1)|x(k)\right) = \left(1-\frac{1}{n_a}\right)\overline{x_a}^2(k) + \frac{1}{4n_a^2}\left(\overline{x_a^2}(k) + \overline{x_b^2}(k)\right) + \left(\frac{1}{n_a} - \frac{1}{2n_a^2}\right)\overline{x_a}(k), \overline{x_b}(k)$$

which gives the first line of Equation (2.1). Applying the same reasoning with partition $\mathcal{V}_b$ gives the second line of Equation (2.1).

For the third and the fourth lines,

$$\begin{aligned}\overline{x_a^2}(k+1) &= \overline{x_a^2}(k) + \frac{1}{n_a}\left(\left(\frac{x_{i_a}(k)+x_{i_b}(k)}{2}\right)^2 - x_{i_a}^2(k)\right)\\ &= \overline{x_a^2}(k) + \frac{1}{n_a}\left(\frac{1}{4}\left(x_{i_a}^2(k)+x_{i_b}^2(k)+2x_{i_a}(k)x_{i_b}(k)\right) - x_{i_a}^2(k)\right).\end{aligned}$$

Now, observe that $\mathbb{E}(x_{i_a}^2(k)|x(k)) = \overline{x_a^2}(k)$, $\mathbb{E}(x_{i_b}^2(k)|x(k)) = \overline{x_b^2}(k)$ and $\mathbb{E}(x_{i_a}(k)x_{i_b}(k)|x(k)) = \overline{x_a}(k)\overline{x_b}(k)$. Taking the expectation with respect to i_a and i_b yields

$$\mathbb{E}(\overline{x_a^2}(k+1)|x(k)) = \frac{4n_a-3}{4n_a}\overline{x_a^2}(k) + \frac{1}{4n_a}\overline{x_b^2}(k) + \frac{1}{2n_a}\overline{x_a}(k)\overline{x_b}(k),$$

which gives the third line of Equation (2.1). Doing the same reasoning for $\overline{x_b^2}(k+1)$ gives the fourth line. Finally, for the fifth line, we compute:

$$\begin{aligned}\overline{x_a}(k+1)\overline{x_b}(k+1) = &\left(\overline{x_a}(k) + \frac{1}{2n_a}\left(x_{i_b}(k)-x_{i_a}(k)\right)\right)\left(\overline{x_b}(k) + \frac{1}{2n_b}\left(x_{i_a}(k)-x_{i_b}(k)\right)\right)\\ = &\ \overline{x_a}(k)\overline{x_b}(k) + \left(\frac{x_{i_a}(k)-x_{i_b}(k)}{2}\right)\left(\frac{\overline{x_a}(k)}{n_b} - \frac{\overline{x_b}(k)}{n_a}\right)\\ &- \frac{1}{4n_an_b}\left(x_{i_a}^2(k) - 2x_{i_a}(k)x_{i_b}(k) + x_{i_b}^2(k)\right).\end{aligned}$$

Taking the conditional expectation gives:

$$\begin{aligned}\mathbb{E}\left(\overline{x_a}(k+1)\overline{x_b}(k+1)|x(k)\right) = &\ \overline{x_a}(k)\overline{x_b}(k)\left(1+\frac{1}{2n_an_b}\left(1-n_a-n_b\right)\right)\\ &+ \frac{\overline{x_a}^2(k)}{2n_b} + \frac{\overline{x_b}^2(k)}{2n_a} - \frac{\overline{x_a^2}(k)+\overline{x_b^2}(k)}{4n_an_b},\end{aligned}$$

which gives the last line of (2.1). □

2.1.2 Replacement

The main difference between a replacement in a complete bipartite graph and in a complete graph is that we have to divide our matrix A_r in two parts. The first one is when the replacement occurs in partition $\mathcal{V}_a$ and the second one when it is in $\mathcal{V}_b$. We have:

$$A_{r,\mathcal{V}_a} = \begin{pmatrix} \frac{n_a-2}{n_a} & 0 & \frac{1}{n_a^2} & 0 & 0\\ 0 & 1 & 0 & 0 & 0\\ 0 & 0 & \frac{n_a-1}{n_a} & 0 & 0\\ 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & \frac{n_a-1}{n_a}\end{pmatrix} \text{ and } A_{r,\mathcal{V}_b} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0\\ 0 & \frac{n_b-2}{n_b} & 0 & \frac{1}{n_b^2} & 0\\ 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & \frac{n_b-1}{n_b} & 0\\ 0 & 0 & 0 & 0 & \frac{n_b-1}{n_b}\end{pmatrix}.$$

When a replacement occurs in the whole system, then we have a probability $\frac{n_a}{n_a+n_b}$ that it affects the partition $\mathcal{V}_a$ and $\frac{n_b}{n_a+n_b}$ that it affects the partition $\mathcal{V}_b$.

Proposition 2.1.2. *The steady state equation (Definition 1.0.4) for a replacement in a complete bipartite graph at time k is given by:*

$$\mathbb{E}X(k+1) = A_r\mathbb{E}X(k), \textit{ where } A_r = \begin{pmatrix} \frac{n_a+n_b-2}{n_a+n_b} & 0 & \frac{1}{n_a(n_a+n_b)} & 0 & 0\\ 0 & \frac{n_a+n_b-2}{n_a+n_b} & 0 & \frac{1}{n_b(n_a+n_b)} & 0\\ 0 & 0 & \frac{n_a+n_b-1}{n_a+n_b} & 0 & 0\\ 0 & 0 & 0 & \frac{n_a+n_b-1}{n_a+n_b} & 0\\ 0 & 0 & 0 & 0 & 1-\frac{2}{n_a+n_b}\end{pmatrix}. \tag{2.2}$$

Proof. Here the only thing we have to compute is $\overline{x_a}(k+1)\overline{x_b}(k+1)$ since all the other elements are provided in Section 1.1.2. If a replacement occurs in partition $\mathcal{V}_a$, and the value of agent i_a is changed, then we obtain:

$$\overline{x_a}(k+1) = \overline{x_a}(k) + \frac{x_{i_a}(k+1)}{n_a} - \frac{x_{i_a}(k)}{n_a}.$$

Taking the expectation and knowing that $\mathbb{E}\left(x_{i_a}(k+1)\right) = 0$ by definition of the distribution that the value held by each agent of the system, we have:

$$\begin{aligned}\mathbb{E}(\overline{x_a}(k+1)|x(k)) &= \overline{x_a}(k) + 0 - \frac{1}{n_a}\overline{x_a}(k)\\ &= \frac{n_a - 1}{n_a}\overline{x_a}(k).\end{aligned}$$

Applying the same reasoning with partition $\mathcal{V}_b$ gives us the results presented in $A_{r,\mathcal{V}_a}$ and $A_{r,\mathcal{V}_b}$ From these two matrices, we know that:

$$A_r = \frac{n_a}{n_a+n_b}A_{r,\mathcal{V}_a} + \frac{n_b}{n_a+n_b}A_{r,\mathcal{V}_b},$$

which proves (2.2). $\square$

And with the same reasoning as in Subsection 1.1.2 but by dividing for each partition, we can obtain vector b_r

$$b_r = \begin{pmatrix} \frac{\sigma^2}{n_a(n_a+n_b)} \\ \frac{\sigma^2}{n_b(n_a+n_b)} \\ \frac{\sigma^2}{n_a+n_b} \\ \frac{\sigma^2}{n_a+n_b} \\ 0 \end{pmatrix}. \tag{2.3}$$

2.1.3 Gossip and replacement

Suppose that an event occurs between time k and $k+1$. This event is the replacement of an agent with probability p or a gossip with probability $1-p$. Denote $x(k)$ the state before the event, $x(k+1)$ the state after. Then, there holds:

$$\mathbb{E}X(k+1) = A\mathbb{E}X(k) + pb_r, \tag{2.4}$$

where b_r is defined in Equation (2.3) and $A = (1-p)A_g + pA_r$. Note that A_g comes from Equation (2.1) and A_r comes from Equation (2.2).

2.1.4 Fixed point

We will now be interested in the study of the fixed point for the steady state equation (2.4) of the complete bipartite graphs where gossips occur with probability $(1-p)$ and replacements occur with probability p . In our case, computing the fixed point is complicated since we are working with vector:

$$X(k) = \begin{pmatrix}\overline{x_a}^2(k) & \overline{x_b}^2(k) & \overline{x_a^2}(k) & \overline{x_b^2}(k) & \overline{x_a}(k)\overline{x_b}(k)\end{pmatrix}^\top,$$

and thus, we have to solve a system made of five equations and five unknowns. Moreover, trying to find the exact solution for the fixed point of complete bipartite graphs seems to be a wrong idea. Indeed, we have tried to compute it using `Mathematica` but the obtained result was not interesting since it looks like a high-degree polynomial approximation.

For this reason, we will no longer be looking for an exact formulation but for numerical values following some parameters. We worked with a system made of $n = 250$ agents where each agent of a partition is connected to all the nodes of the other partition, *i.e.*, with a complete bipartite graph of 250 agents. As we did in the previous sections, the values held by the agents follow a uniform distribution over $[-\frac{1}{2}, \frac{1}{2}]$ with a zero mean such that $\sigma^2 \approx \frac{1}{12}$ (see Appendix A). The two parameters that will change during our simulations will be the probability of replacement $p = 0.05 + 0.10i \quad \forall i = 0, 1, \ldots, 9$ and the size of the first partition $n_a = 5 + 10j \quad \forall j = 0, 1, \ldots, 24$.

In Figures 2.2 and 2.3 we can see the evolution of the fixed point of the square mean value of $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively. Since we are working with a complete bipartite graph, we can observe a symmetry around $n_a = 125$ if we compare the two figures. This is to say that results when $n_a = x$ in Figure 2.2 are equal to results when $n_a = 250 - x$ in Figure 2.3, where $x \in [1, 249] \cap \mathbb{N}$. We observe that the fixed point grows when the considered partition becomes very small compared to the size of the system. This growth can be accentuated with high probability of replacement. These results fit with our expectation since if we are considering a big partition, the mean of the value of each agent will tend towards 0. On the other hand, if this partition is small, then it will become more sensible, with the consequence of increasing its square mean value. Since we are working with a graph relatively large, when $p = 0$, the fixed point tends towards zero due to the distribution of the values held by the agents.

Figures 2.4 and 2.5 show the fixed point of the mean square value of $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively. On the contrary of the square mean value, the more the partition we consider is large, the more its mean square value is important. We can explain this phenomenon since the mean square value is the sum of the square mean value and the variance. For a big partition, the square mean value will tend towards 0 while the variance will tend towards σ^2 multiplied by a constant. On the other hand, for a very small partition, the variance will tend towards 0 (see Appendix A). However, when p growths, the mean square value increases too.

In Figures 2.6 and 2.7 we see the fixed point of the variance of $\mathcal{V}_a$ and $\mathcal{V}_b$ respectively. Their variance is equal to the mean square value minus the square mean value. Thus, these results are not surprising. Intuitively, we could guess that the bigger p, the larger the variance is. Indeed, if there is no replacement, after some time steps, all the agents will have the same value and then the variance is equal to 0. On the other hand, if $p = 1$, then the fixed point of our variance will tend towards $\frac{\sigma^2}{n_a}$ is we consider partition $\mathcal{V}_a$ and $\frac{\sigma^2}{n_b}$ is we consider partition $\mathcal{V}_b$.

Finally, Figure 2.8 shows the evolution of the fixed point of $\overline{x_a x_b}$. If $p = 1$, at each time step, a replacement occurs. Looking at the replacement matrix (Equation (2.2)), we see that it tends to decrease the value of $\overline{x_a x_b}$. We can explain this phenomenon by saying that when only replacements occur, the two partitions become less and less correlated. When $p \in (0, 1)$, we can use the following reasoning: the more there are replacements, the more $\overline{x_a}$ and $\overline{x_b}$ are decorrelated by definition of the replacement matrix, so when p grows, their product has to decrease. However, we can observe that when p is equal to 0, the fixed point is smaller than $p = 0^+$. Not only because when $p = 0^+$ the decorrelation is very weak, but also since we have to add to our fixed point a fraction of σ^2. Now we will analyze how the fixed point behaves following the size of our partitions. As we have seen for the fixed point of the square mean value, when a partition is very small, the fixed point of its square mean value is the largest. For this reason, it is logical that the fixed point for $\overline{x_a x_b}$ is the largest when one of the two partitions is very small. Moreover, the parity around $n_a = 125$ is logical since we are working with a complete bipartite graph. Therefore, when the two partitions have the same size, we must have an optimum. This one is a minimum since the mean of the partition tends towards 0 when the size of this partition growths. We can also add that the gap between the mean of two consecutive sizes of partition becomes smaller and smaller when the size of the partition increases. Thus, the gap between n_a and $n_a + 1$ is greater than the one between $n_a + 1$ and $n_a + 2$. For these reasons, when $n_a = 125$, we obtain a minimum value for a given p.

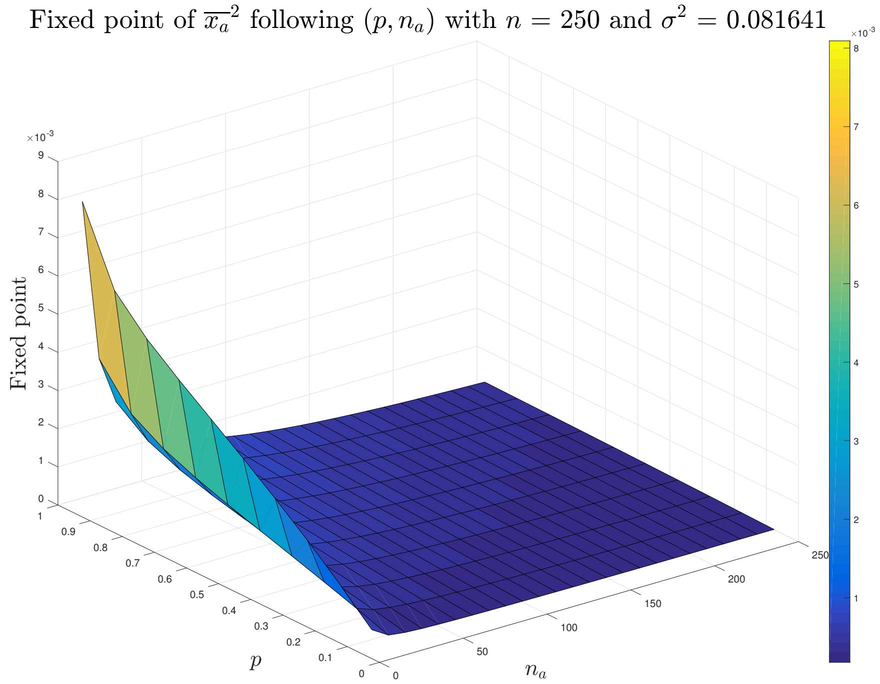


Figure 2.2: Fixed point of the square mean value of $\mathcal{V}_a$ following (p, n_a) using Equation (2.4).

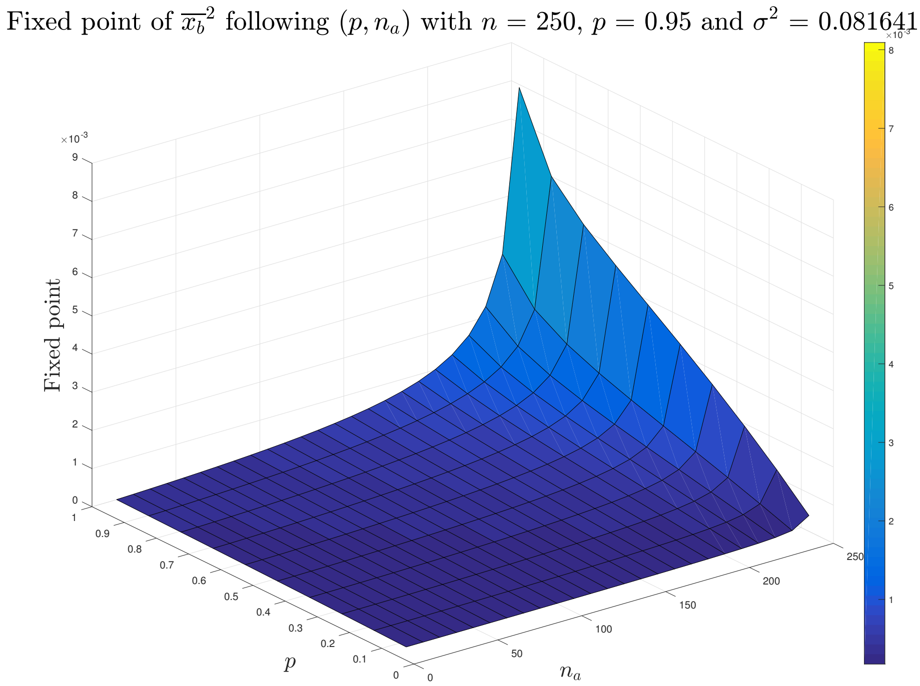


Figure 2.3: Fixed point of the square mean value of $\mathcal{V}_b$ following (p, n_a) using Equation (2.4).

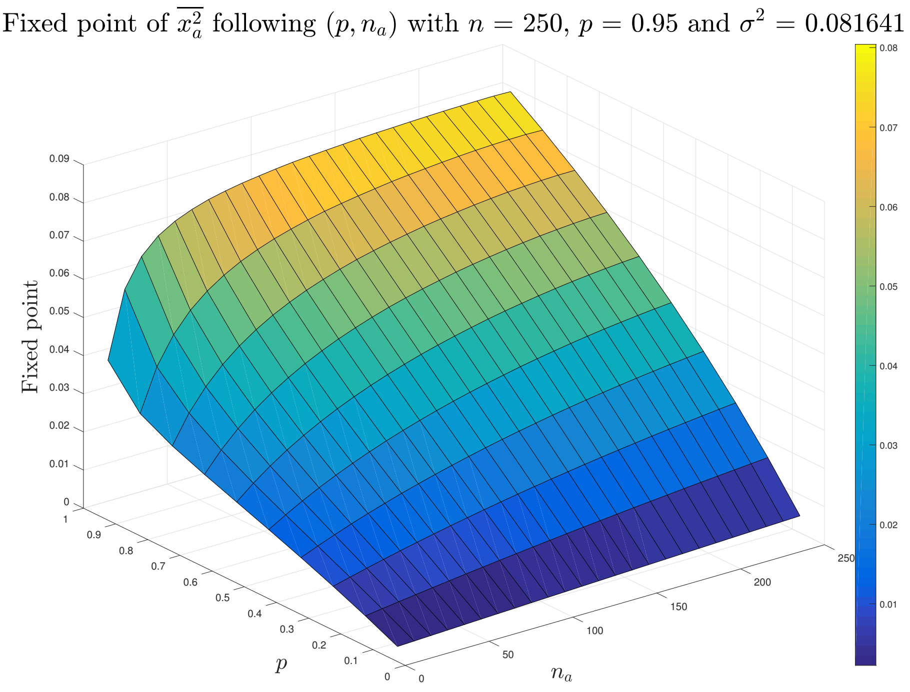


Figure 2.4: Fixed point of the mean square value of $\mathcal{V}_a$ following (p, n_a) using Equation (2.4).

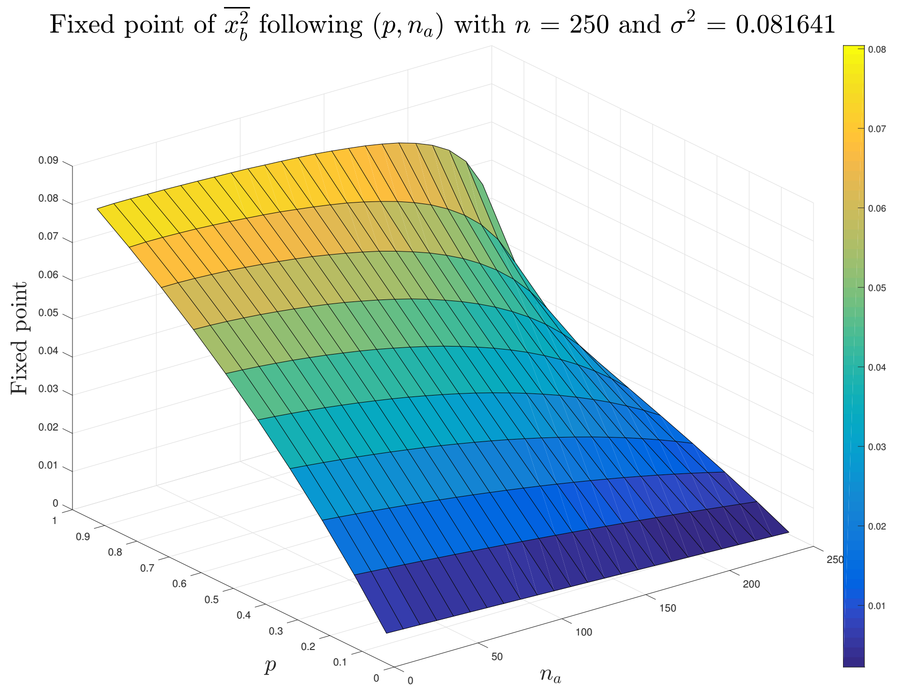


Figure 2.5: Fixed point of the mean square value of $\mathcal{V}_b$ following (p, n_a) using Equation (2.4).

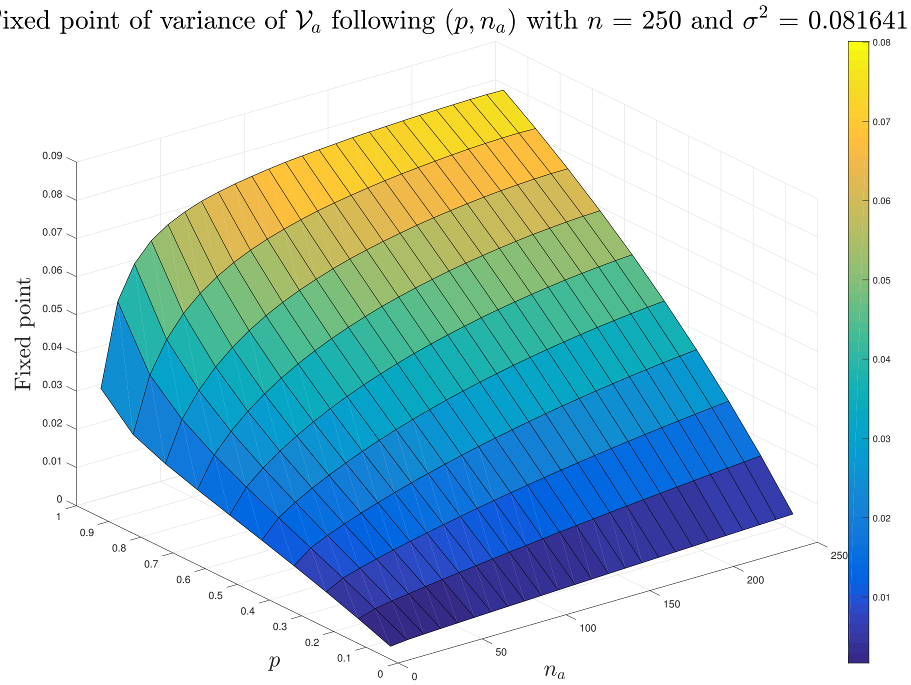


Figure 2.6: Fixed point of variance of $\mathcal{V}_a$ following (p, n_a) using Equation (2.4).

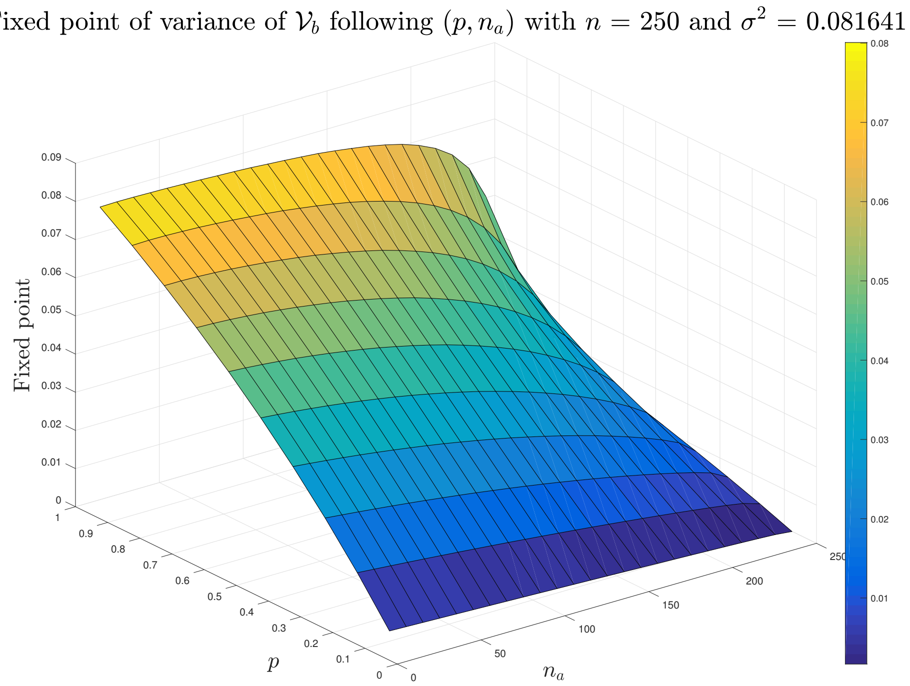


Figure 2.7: Fixed point of variance of $\mathcal{V}_b$ following (p, n_a) using Equation (2.4).

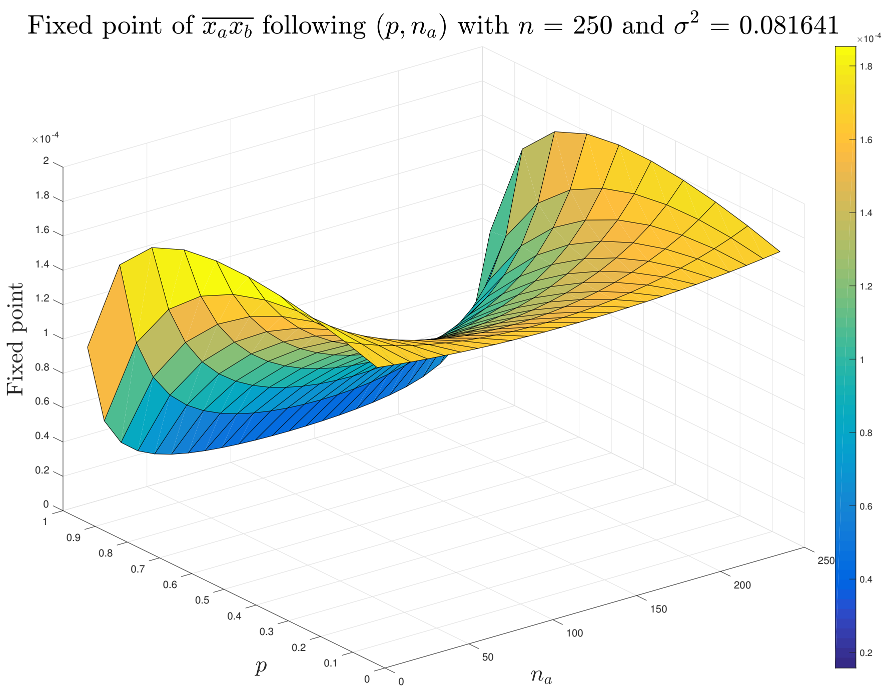


Figure 2.8: Fixed point of $\overline{x_a x_b}$ following (p, n_a) using Equation (2.4).

2.1.5 Variance of the whole graph

Now we want to obtain the variance of the whole graph given information about the two partitions. Working on the entire system will allow us to compare the numerical values of our fixed point between complete bipartite graphs and complete graphs. We know that at each time k with $k \geq 0$, we have:

$$\begin{aligned}
\overline{x}(k) &= \frac{1}{n}\sum_{i=1}^{n} x_i(k) = \frac{1}{n}\left(\sum_{i_a=1}^{n_a} x_{i_a}(k) + \sum_{i_b=1}^{n_b} x_{i_b}(k)\right) \\
&= \frac{n_a\overline{x_a}(k) + n_b\overline{x_b}(k)}{n_a + n_b}, \\
\overline{x^2}(k) &= \frac{1}{n}\sum_{i=1}^{n} x_i^2(k) = \frac{1}{n}\left(\sum_{i_a=1}^{n_a} x_{i_a}^2(k) + \sum_{i_b=1}^{n_b} x_{i_b}^2(k)\right) \\
&= \frac{n_a\overline{x_a^2}(k) + n_b\overline{x_b^2}(k)}{n_a + n_b}.
\end{aligned}$$

And we know that $\mathrm{Var}(x(k)) = \overline{x^2}(k) - \overline{x}^2(k)$.

Proposition 2.1.3. *The variance of a complete bipartite graph given the characteristics of its two partitions is given by:*

$$\mathrm{Var}(x(k)) = \frac{n_a^2\mathrm{Var}(x_a(k)) + n_b^2\mathrm{Var}(x_b(k)) + n_a n_b(\overline{x_a^2}(k) + \overline{x_b^2}(k) - 2\overline{x_a}(k)\overline{x_b}(k))}{(n_a+n_b)^2}.$$

Proof. By definition of the variance, it follows that:

$$\begin{aligned}
\mathrm{Var}(x(k)) &= \frac{n_a\overline{x_a^2}(k) + n_b\overline{x_b^2}(k)}{n_a+n_b} - \frac{n_a^2\overline{x_a}^2(k) + 2n_an_b\overline{x_a}(k)\overline{x_b}(k) + n_b^2\overline{x_b^2}(k)}{(n_a+n_b)^2} \\
&= \frac{(n_a^2+n_an_b)\overline{x_a^2}(k) + (n_b^2+n_an_b)\overline{x_b^2}(k) - (n_a^2\overline{x_a}^2(k) + 2n_an_b\overline{x_a}(k)\overline{x_b}(k) + n_b^2\overline{x_b}^2(k))}{(n_a+n_b)^2} \\
&= \frac{n_a^2\mathrm{Var}(x_a(k)) + n_b^2\mathrm{Var}(x_b(k)) + n_an_b(\overline{x_a^2}(k) + \overline{x_b^2}(k) - 2\overline{x_a}(k)\overline{x_b}(k))}{(n_a+n_b)^2}.
\end{aligned}$$

□

Now, if we work with a system where $n = 250$ and if we compare the difference between the fixed point of the variance of the complete bipartite graph and the fixed point of the variance of the complete graph we obtain the results presented in Figure 2.9. Note that in this context, a gossip is between two agents so that we have the following fixed point for the variance of the complete graph [HM17]:

$$\mathrm{Var}(x)|_{\mathrm{eq}} = \frac{2p(n-1)}{p+2n-1}\sigma^2, \tag{2.5}$$

where p is as usual the probability for a replacement at each time step. We observe that whatever the couple of values for (p, n_a), the variance of the complete bipartite graph is always greater than the one of the complete graph with the same number of agents. This phenomenon will be developed in the third chapter. We also observe that the gap between the two variances is greater when a partition is very small or very large compared to n. Results when p varies ensue from the behaviour of the fixed point of $\mathrm{Var}(x_a)$, $\mathrm{Var}(x_b)$, $\overline{x_a^2}$, $\overline{x_b^2}$ and $\overline{x_a x_b}$.

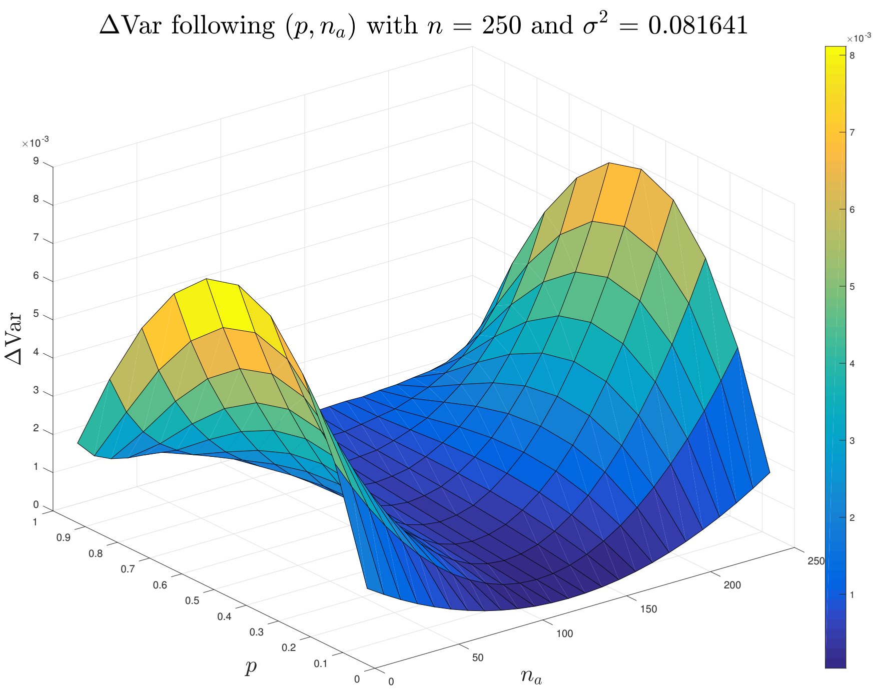


Figure 2.9: $\Delta\mathrm{Var}(x)$ following (p, n_a) using Equation (2.4). The difference is between the variance of the agents of a complete bipartite graph and the variance of the agents of a complete graph with the same number of agents and the same probability of replacement p.

2.1.6 Simulations

In this section we will implement the steady state equation for complete bipartite graphs (Equation (2.4)), obtained in Subsection 2.1.3. First of all, we will be focused on a graph made of 250 agents with two equally sized partitions ($n_a = 125$ and $n_b = 125$). The value associated to each agent is uniformly distributed on $[-\frac{1}{2}, \frac{1}{2}]$ with mean 0 and so the associated variance is $\sigma^2 \approx \frac{1}{12}$ (see Appendix A). The probability that a replacement occurs is equal to $p = 0.05$.

To begin, we will work with simulations taking place over 50 000 time steps. In Figures 2.10 and 2.11, we can see the square mean value of $\mathcal{V}_a$ and $\mathcal{V}_b$ compared to their expected value. Since the square mean value oscillates very quickly, we also draw the time-average (Definition 1.1.2) of our square mean value. Note that the average tends towards the expectation for both partitions. In Figures 2.12 and 2.13, we observe that our variance fits pretty well with the expectation. It is due to the fact that we redefine σ^2 in our simulation in order to use a value that fits the best with the average σ^2 for partitions of this size (see Appendix A). We can see that for both partitions, our results are oscillating around the expected behaviour while the time-average converges towards the expected value.

Finally, in Figure 2.14, the evolution of $\overline{x_a}(k)\overline{x_b}(k)$ is plotted. As we can see the product of both means is quite near to 0. It would be nearer 0 if the probability p was bigger. Indeed, as shown previously, the replacement matrix A_r defined in (2.2) aims to decorrelate the mean of both partitions.

2.2 Star Graph

To finish this chapter, we will study the particular case of the star graph. It has the particularity that the first partition contains only one agent that is connected to all the other agents while the second partition is of size $n-1$. First of all, we will analyze how this sort of systems behaves and what does its steady state equation looks like. Secondly, we will try to answer to the following question: *which node to take into account in order to have a good approximation of the behaviour of the graph?* The star graph with 8 nodes has been drawn in Figure 2.15.

From Equation (2.1), we have:

Proposition 2.2.1. *The steady state equation (Definition 1.0.4) of the star graph with $n_a = 1$ and $n_b = n-1$ when only gossips occur is given by:*

$$\mathbb{E}X(k+1) = A_g \mathbb{E}X(k), \text{ where } A_g = \begin{pmatrix} 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ 0 & 1-\frac{1}{n-1} & \frac{1}{4(n-1)^2} & \frac{1}{4(n-1)^2} & \frac{2n-3}{2(n-1)^2} \\ 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ 0 & 0 & \frac{1}{4(n-1)} & 1-\frac{3}{4(n-1)} & \frac{1}{2(n-1)} \\ \frac{1}{2(n-1)} & \frac{1}{2} & \frac{-1}{4(n-1)} & \frac{-1}{4(n-1)} & \frac{1}{2} \end{pmatrix}. \quad (2.6)$$

From Equation (2.2), we have:

Proposition 2.2.2. *The steady state equation (Definition 1.0.4) for a replacement in a star graph is given by with $n_a = 1$ and $n_b = n-1$:*

$$\mathbb{E}X(k+1) = A_r \mathbb{E}X(k), \text{ where } A_r = \begin{pmatrix} \frac{n-3}{n} & 0 & \frac{1}{n} & 0 & 0 \\ 0 & \frac{n-3}{n} & 0 & \frac{1}{n(n-1)} & 0 \\ 0 & 0 & \frac{n-1}{n} & 0 & 0 \\ 0 & 0 & 0 & \frac{n-1}{n} & 0 \\ 0 & 0 & 0 & 0 & 1-\frac{2}{n} \end{pmatrix}. \quad (2.7)$$

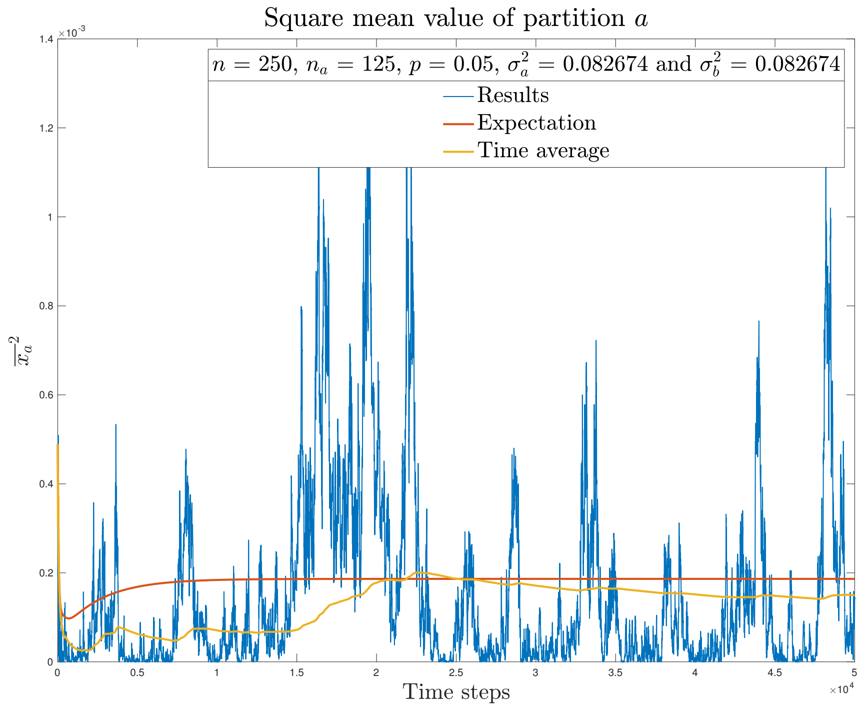


Figure 2.10: Square mean value of partition $\mathcal{V}_a$ for a bipartite graph with 50 000 time steps using Equation (2.4).

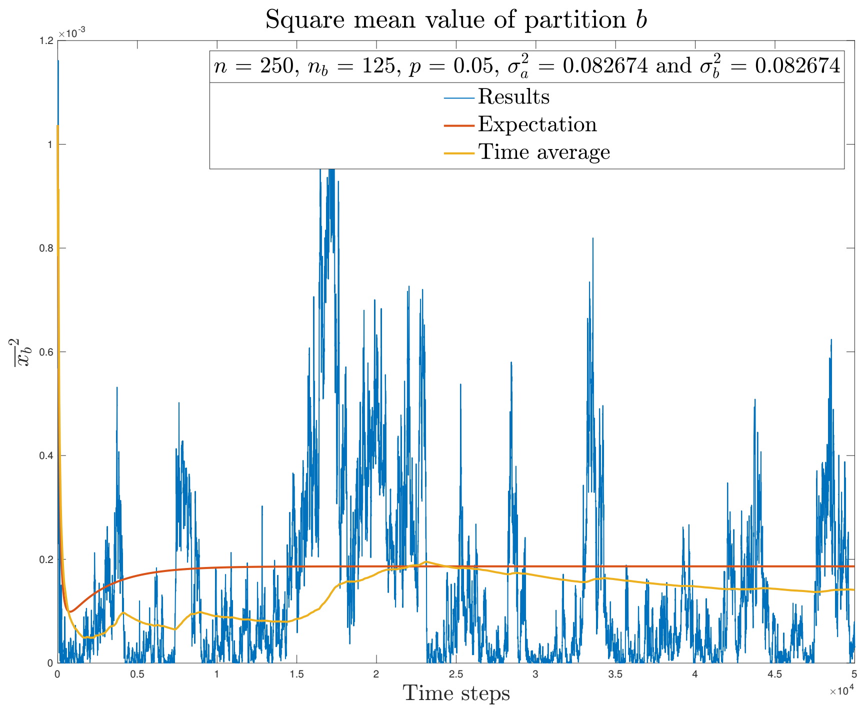


Figure 2.11: Square mean value of partition $\mathcal{V}_b$ for a bipartite graph with 50 000 time steps using Equation (2.4).

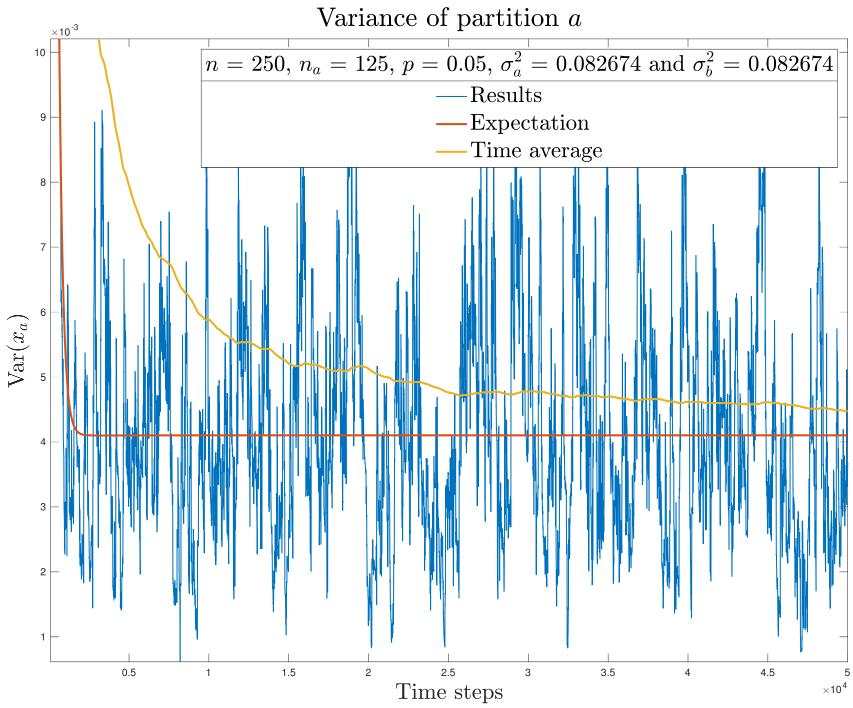


Figure 2.12: Variance of partition $\mathcal{V}_a$ of a bipartite graph with 50 000 time steps using Equation (2.4).

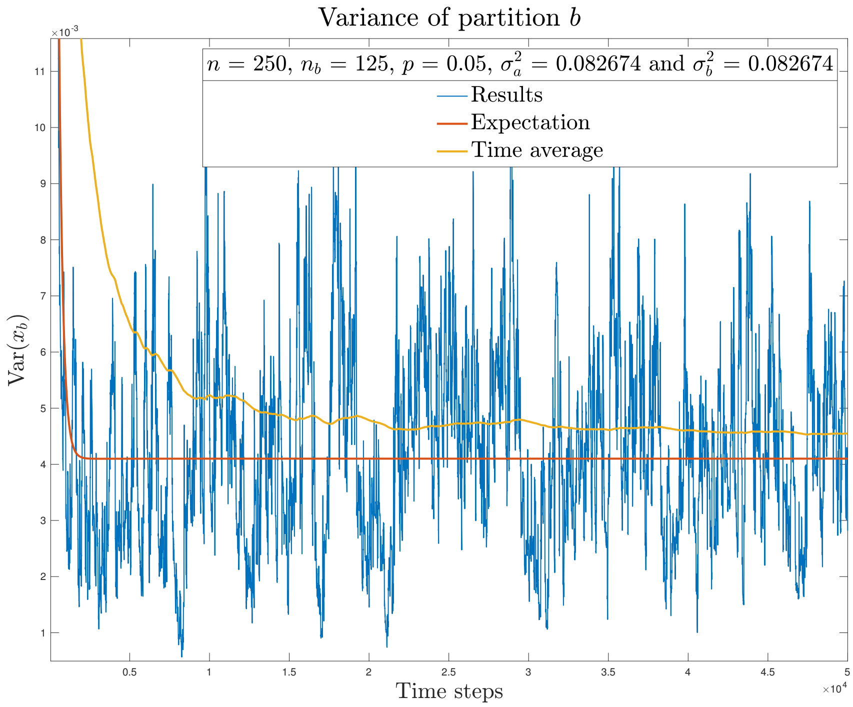


Figure 2.13: Variance of partition $\mathcal{V}_b$ of a bipartite graph with 50 000 time steps using Equation (2.4).

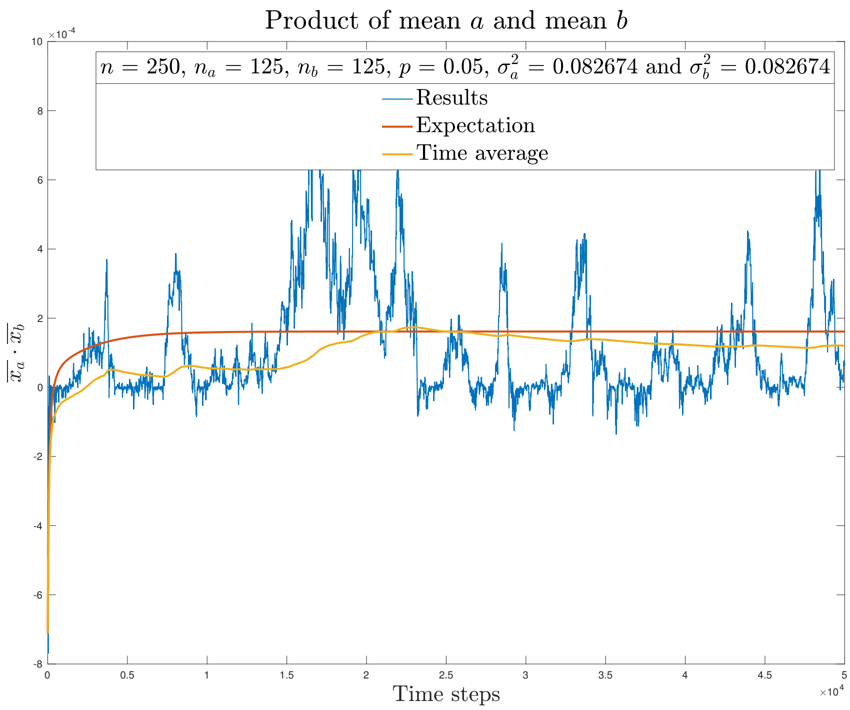


Figure 2.14: Evolution of $\overline{x_a}(k)\overline{x_b}(k)$ using Equation (2.4).

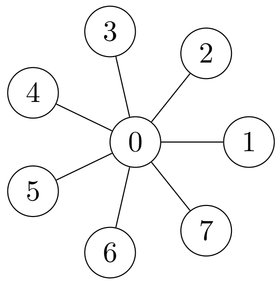


Figure 2.15: The star graph with 8 nodes.

If a replacement occurs with probability p and a gossip occurs with probability $1-p$, then the steady state equation associated is the sum of Equation (2.6) multiplied by $1-p$ and Equation (2.7) multiplied by p.

2.2.1 Simulations

We will still be focused on a graph made of 250 agents. The value associated to each agent is uniformly distributed on $[-\frac{1}{2}, \frac{1}{2}]$ and so the associated variance is $\sigma^2 \approx \frac{1}{12}$ (see Appendix A). The probability that a replacement occurs is equal to $p = 0.05$. $\mathcal{V}_a$ is the partition containing a single agent.

In Figures 2.16 and 2.17, we compare values of the expected square mean value, the square mean value and the time-average of the square mean value (Definition 1.1.2). For partition $\mathcal{V}_a$, this is to say for the partition containing only one agent, we only display the expected square mean value and the time-average of the square mean value. This choice has been motivated by the fact that, at each time step, x_{i_a} is either implied in a gossip or is replaced. In both cases, $\overline{x_a}(k)$ changes significantly, so that it does not make sense to display it since it will change the scale of our graph, and we will obtain results whose interpretation is harder. Beside that, we can say that our results fit very well with the expected one.

Figures 2.19 and 2.20 show the variance of partitions $\mathcal{V}_a$ and $\mathcal{V}_b$. For partition $\mathcal{V}_a$, it is obvious that its variance will be equal to 0 since it contains one and only one agent. To explain that the expectation is not always equal to zero, we suggest that there are some numerical errors of order 10^{-18}. For partition $\mathcal{V}_b$, results fit with the expectation.

Finally, in Figure 2.22, we show the time-average of $\overline{x_a}(k)\overline{x_b}(k)$ and its expected behaviour from Equations (2.6) and (2.7). For the same reason as for the square mean value of partition $\mathcal{V}_a$, we will be focused on the time-average of the square mean value and not only on the obtained square mean value since it changes very quickly and disturbs the scale of our graph.

2.2.2 Center versus leaves for approximating star graph behaviour

Now we will try to determine which agent is the best in order to approximate the average behaviour of the star graph. By definition of the topology of this sort of graphs, we only have two possibilities: either the central node, *i.e.*, the one that is connected to all the other agents, or a node randomly selected in the partition of size $n-1$. We say that we can randomly choose this node since all the agents of this partition have the same position in the graph.

By behaviour, we mean the average of the values of the graph. The graph we consider is made of 250 agents. In Figure 2.23, we observe the behaviour of the central node in light blue compared to the one of a randomly selected node in dotted red. Obviously, since the central node is implied in every gossip, its value oscillates at each time steps. Moreover, we can say that when the red curve changes its value, it holds the same number than the central node. As we can see, it is difficult to interpret something from this plot. For this reason, we will analyze the time-average (Definition 1.1.2) of both the central node and a randomly selected node. In Figure 2.24, we plot the same results as in Figure 2.23 but taking the time-average of the central node and the other node. In this graph, it is easier to visualize the behaviour of our agents. We can clearly see that the central node, despite its frequent oscillations, approximates better the average behaviour of the graph.

We can thus complete this section with the following conclusion: the value held by the central node is not a very interesting approximation of the behaviour of the graph since it oscillates too much. On the other hand, the time-average of this node follows quite well the average of the graph. We can thus use the time-average of the central node in other to have an idea of the behaviour of our system. This information is quite interesting since it means that we can simplify a graph, potentially huge, to a single node.

2.3 Conclusion

Through this chapter, we studied how open multi-agent systems with a complete bipartite graph structure behave. First of all, it is possible to obtain a steady state equation for such graphs. However, the fixed points associated are not compact as they are for complete graphs. For this reason, we had to evaluate them as functions of p and the size of the partition n_a.

Secondly, we discovered that the fixed point of the variance of the complete graph is always smaller than the one of any bipartite graph for the same number of agents, whatever p and n_a are. This point will be the motivation for the third chapter, in which we will explain why the variance behaves in this way.

Finally, we also investigated in much details on a particular case of bipartite graph, called the star graph, where the first partition is made of a singular agent while the second one contains all the other agents. We have learned from this section that *it is possible to approximate the*

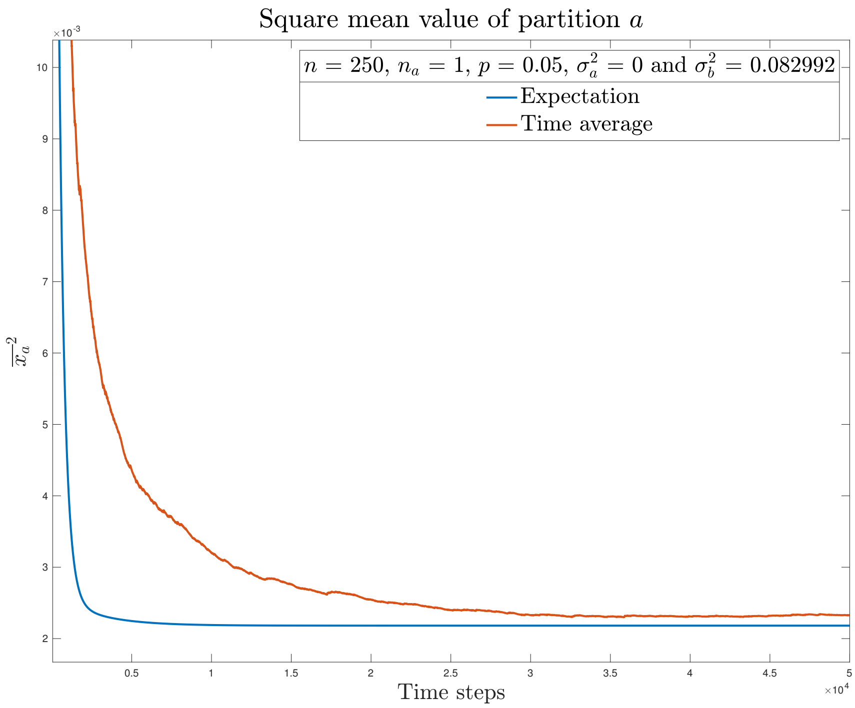


Figure 2.16: Square mean value of partition a.

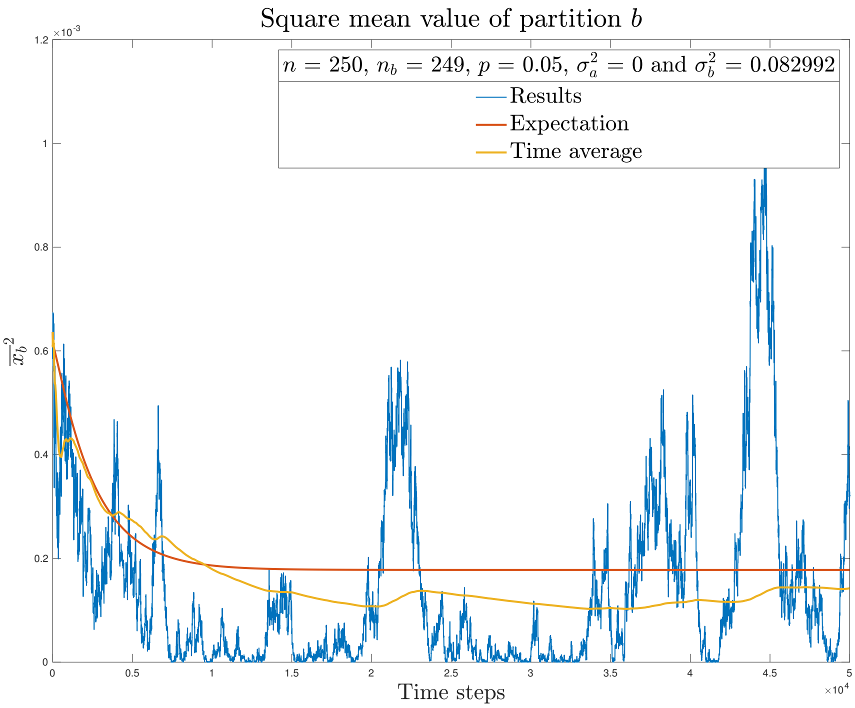


Figure 2.17: Square mean value of partition b.

Figure 2.18: Square mean value for a star graph with 50 000 time steps using Equations (2.6) and (2.7).

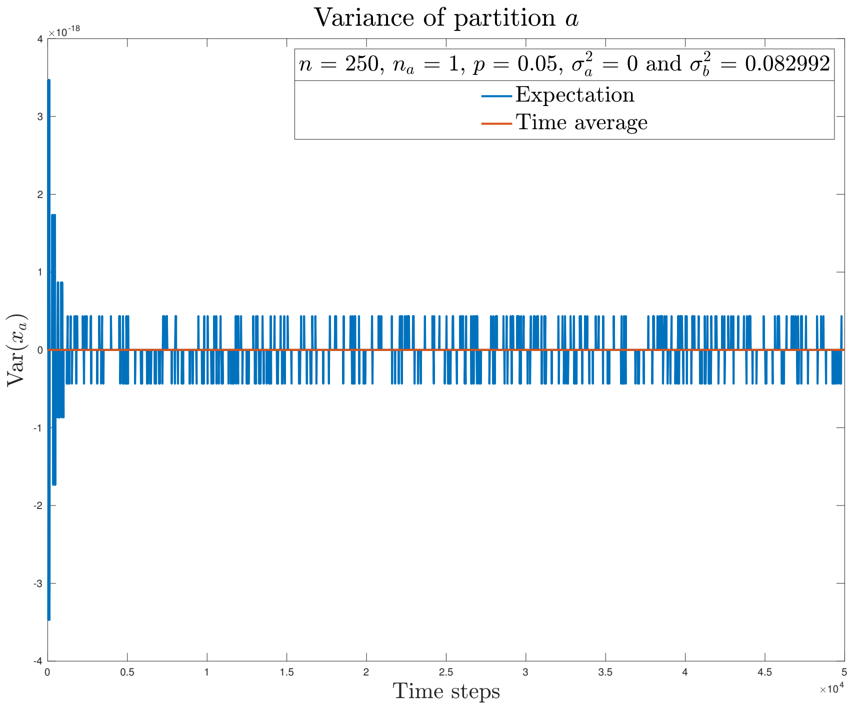


Figure 2.19: Variance of partition a.

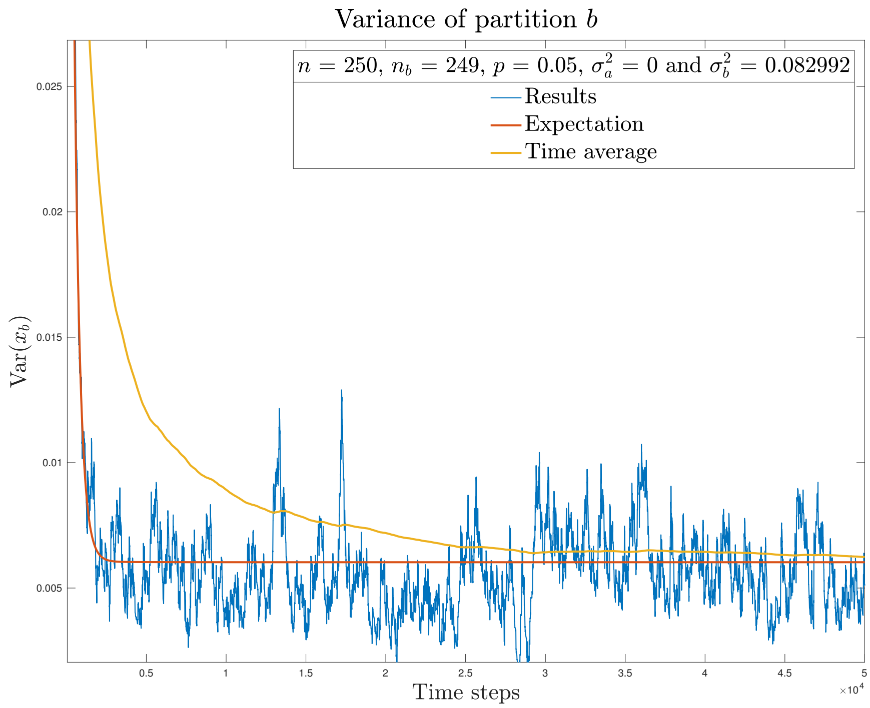


Figure 2.20: Variance of partition b.

Figure 2.21: Variance for a star graph with 50 000 time steps using Equations (2.6) and (2.7).

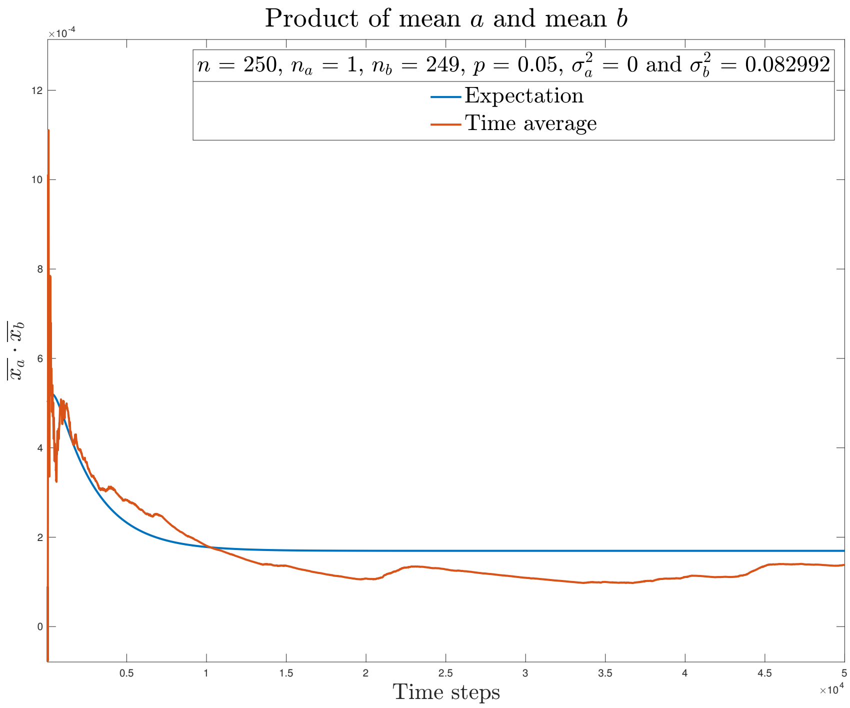


Figure 2.22: Evolution of $\overline{x_a}(k)\overline{x_b}(k)$ for a star graph with 50 000 time steps using Equations (2.6) and (2.7).

behaviour of the system with only the central node. This point is very important since it means that we can resume the behaviour of a huge system only by the behaviour of a single agent or a very small partition.

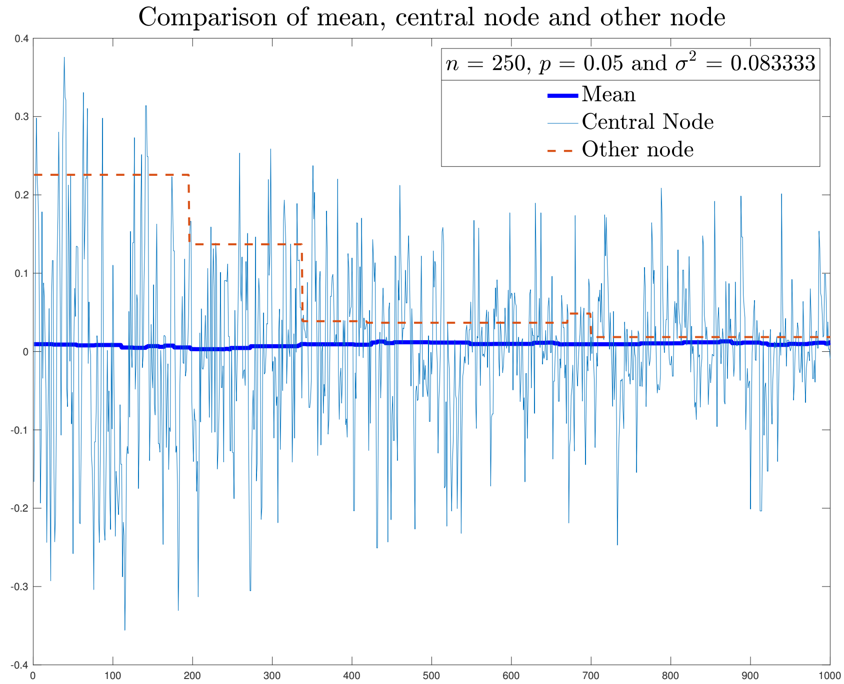


Figure 2.23: Mean of a star graph made of 250 agents and comparison between the behaviour of the central node and a randomly selected node over 1000 time steps.

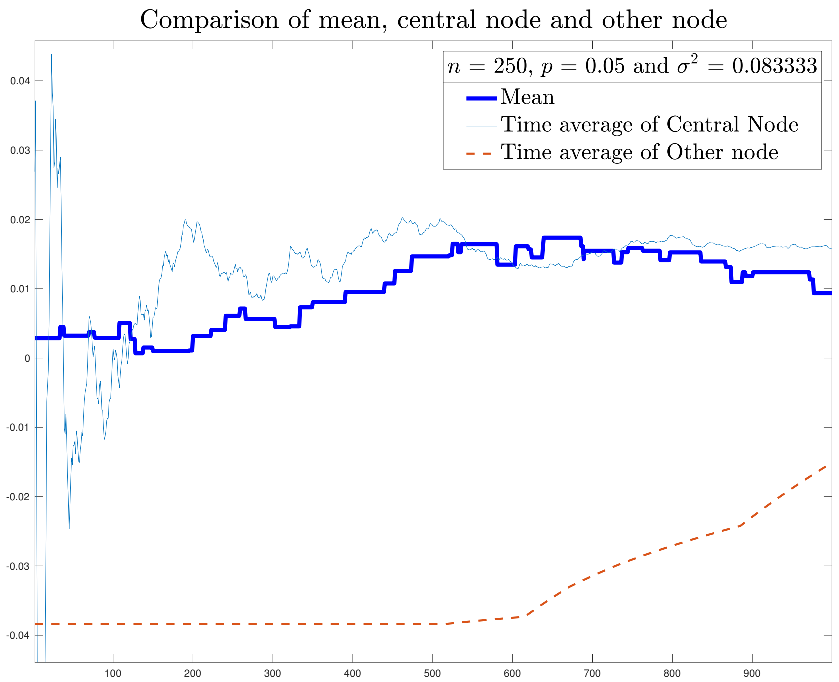


Figure 2.24: Mean of a star graph made of 250 agents and comparison between the time-average (Definition 1.1.2) behaviour of the central node and the time-average of a randomly selected node over 1000 time steps.

Chapter 3

Systems with General Non-Complete Graph Structure

In this chapter we will compare the variance of some graphs to the variance of the complete graph with the same number of agents. As we saw in Figure 2.9, the variance of a bipartite graph is always greater than the one of the complete graph with the same number of agents and the same probability of replacement. This statement is true whatever the sizes of both partitions in the bipartite graph are.

To begin with, we will study a new particular topology of graphs called the barbell graphs (Definition 3.1.1). This structure will be used in order to highlight the dependence of the variance of a system on the parameters of this sort of graphs. Then we will present some evidences abounding in the sens of Conjecture 3.0.1. First of all, we will check whether the steady state equation of the complete graph when two agents are implied in a gossip is also valid for some sorts of network having the vertex-transitive property. Then we will try to derive a link between the diameter of a graph and its variance. Finally, we will adapt some results of [BGPS06] in order to obtain bounds on the variance of a graph as functions of the eigenvalues of its expected transition matrix (Equation (3.2)).

Conjecture 3.0.1. *The variance of any graph $\mathcal{G}$ on which gossips and replacements occur is always greater or equal to the expected one of the complete graph with the same number of vertices as $\mathcal{G}$.*

3.1 Barbell Graph

Definition 3.1.1. *A **barbell graph** is characterized by two complete graphs of size n_2 connected by a single edge. In order to have an additional degree of freedom, we slightly modify its structure, and we add one more parameter, d, defining the number of nodes in the path between the two complete graphs,* i.e.*, the length of the path between the two parts minus one.*

For example, Figure 3.1 shows a barbell graph with 21 nodes, each complete graph made of 10 nodes and $d = 3$.

In Figures 3.2 and 3.3 we respectively observe the evolution of the square mean value and the variance of a barbell graph with $n_2 = 200$ and $d = 30$. For both graphs, the red curve corresponds to the expected behaviour for a complete graph. The time-average (Definition 1.1.2) of the square mean value does not differ from results for a complete graph. However, the variance of the barbell graph will always be above the expected one of the complete graph whatever the number of time steps we consider.

In Figure 3.4 is plotted the gap between the square mean value of a barbell graph and the expected square mean value of the complete graph with the same number of agents and

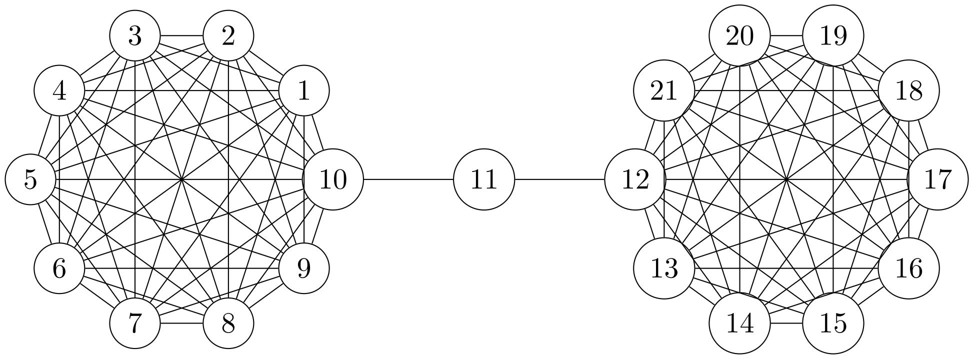


Figure 3.1: The barbell graph with $n_2 = 10$ and $d = 3$.

probability of replacement p, while in Figure 3.5 is plotted the variance of the same barbell graph and the expected variance of the complete graph following the value of d and after 10000 time steps. The square mean value is not affected by this parameter and its value stays more or less constant. On the other hand, the variance of the barbell graph grows linearly with parameter d. It is also possible to observe how the square mean value and the variance of a barbell graph behave with $n_2 = 50$ and $n_2 = 100$. Since they evolve as for $n_2 = 200$, those results are presented in Appendix C. Note that, for all these graphs, we have taken the average behaviour on 10 simulations with always the same parameter. This choice has been motivated by the gaps that are possible from one simulation to another. Doing as we did allows us to have a sort of average behaviour.

We can also wonder how does the system evolve when d is fixed and n_2 changes. The larger n_2, the smaller the variance is for a given d. To be convinced that this property holds, Figures 3.6 and 3.7 describe the evolution of the gap between the expectation for a complete graph and the results obtained for a Barbell graph in average for 6 simulations and a distance $d = 15$. We can clearly conclude that the larger n_2 is, the smaller the square mean value and the variance are. This conclusion is not surprising. Indeed, if we consider a barbell graph made of 25 nodes with $d = 15$, there are not a lot of edges compared to the number of agents. Now, consider a barbell graph made of 1015 nodes with $d = 15$. Proportionally, this second barbell graph will be more connected than the first one and intuitively, it can be expected that this second network has a smaller square mean value and a smaller variance than the first one. Indeed, the diameter of the graph is proportionally smaller compared to its number of agents.

3.2 Vertex-Transitive Graphs

The previous section illustrates the fact that some graphs have a variance that is greater than the one of the complete graph. Now, we will study the behaviour of vertex-transitive graphs. We want to verify whether only the full-mesh property, *i.e.*, the complete graph, implies results we obtained in the first chapter and in [HM17] or if more general structures, such as vertex-transitivity graphs, can also reach the expected behaviour derived from Equation (1.2). We are thus going to make some researches in the context of Conjecture 3.0.1.

Definition 3.2.1. *A **vertex-transitive graph** is a graph $\mathcal{G}$ such that, given any two vertices v_1 and v_2 of $\mathcal{G}$, there is some automorphism*

$$f : \mathcal{V}(\mathcal{G}) \to \mathcal{V}(\mathcal{G}),$$

such that:

$$f(v_1) = v_2.$$

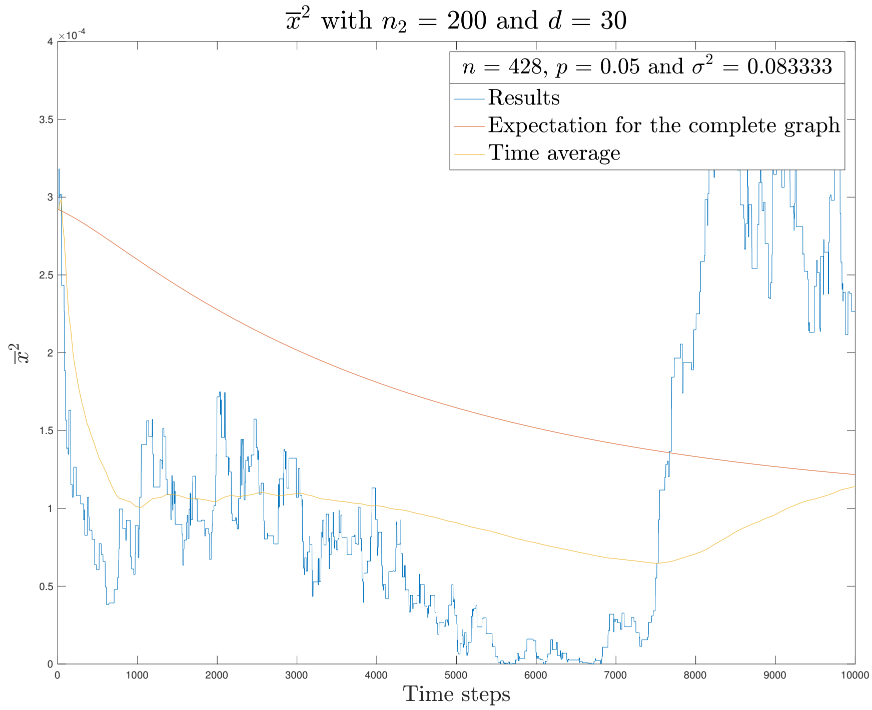


Figure 3.2: In red, the expected square mean value for a complete graph when gossips between 2 agents occur with probability $1 - p$ on 5000 time steps using Equation (1.2). In blue, the true square mean value (see Notation) of a barbell graph with $n_2 = 200$ and $d = 30$ at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

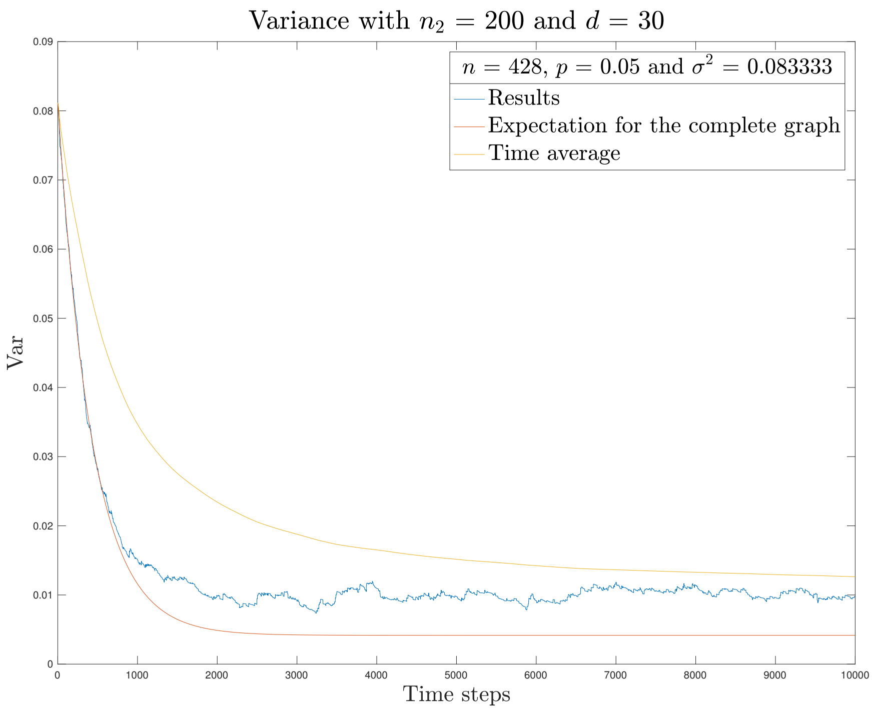


Figure 3.3: In red, the expected variance for a complete graph when gossips between 2 agents occur with probability $1 - p$ on 5000 time steps using Equation (1.2). In blue, the true variance (see Notation) of a barbell graph with $n_2 = 200$ and $d = 30$ at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

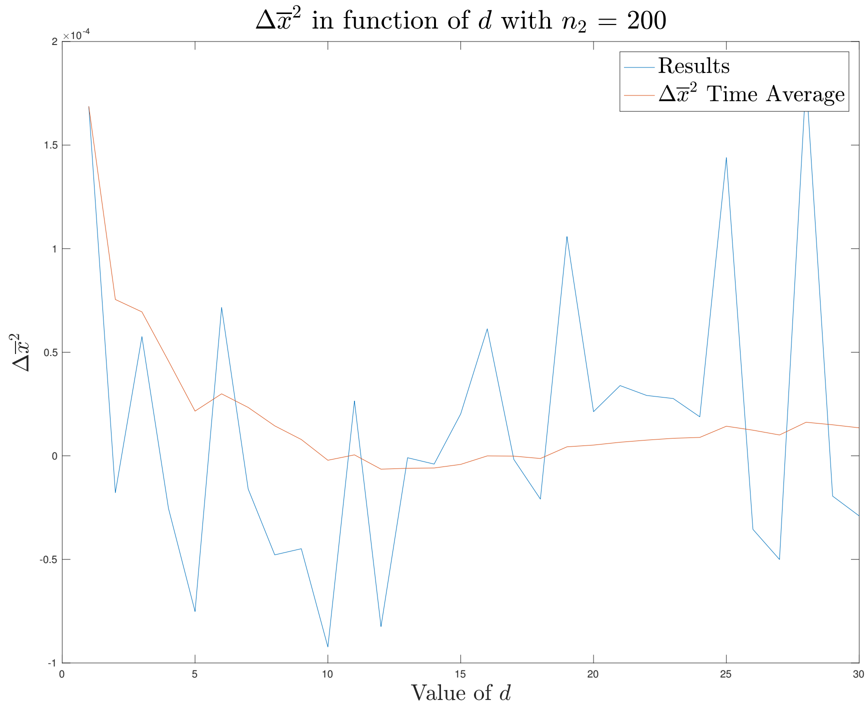


Figure 3.4: In blue, the average gap on 10 simulations between the true square mean value (see Notation) for a barbell graph made of $n_2 = 200$ agents following d and the true square mean value of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

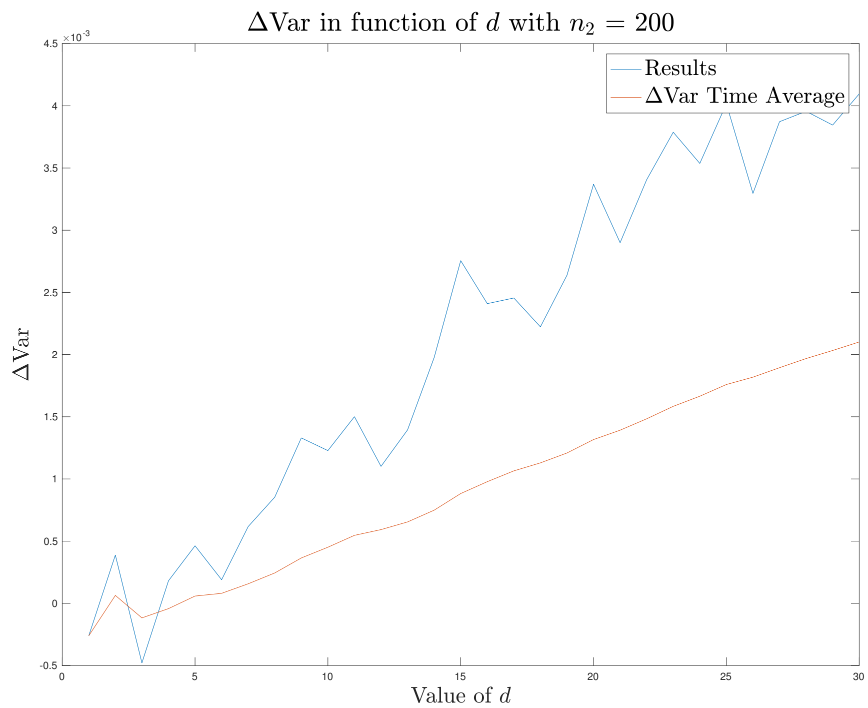


Figure 3.5: In blue, the average gap on 10 simulations between the true variance (see Notation) for a barbell graph made of $n_2 = 200$ agents following d and the true variance of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

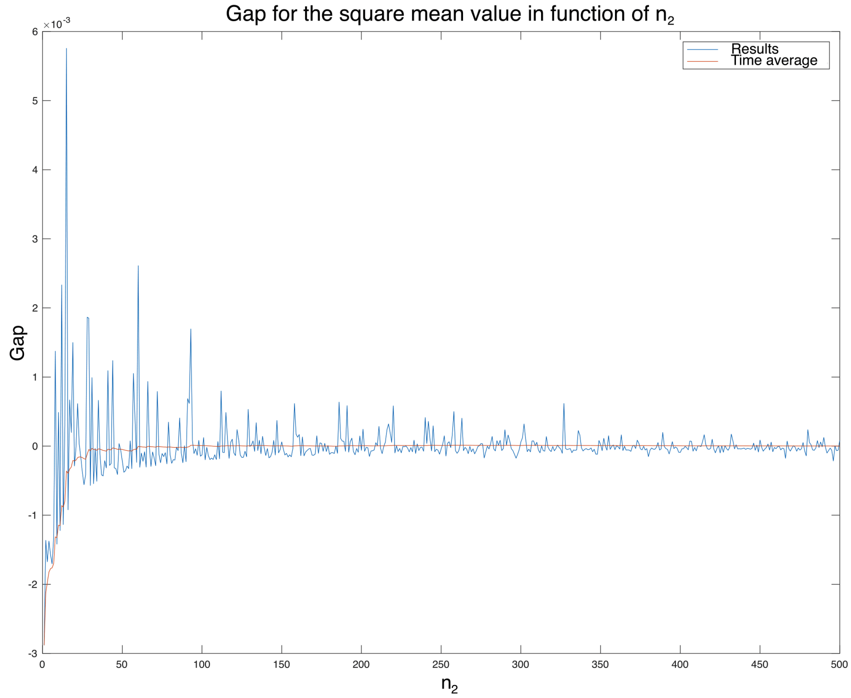


Figure 3.6: In blue, the average gap on 6 simulations between the true square mean value (see Notation) for a barbell graph made of $d = 15$ agents following n_2 and the true square mean value of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

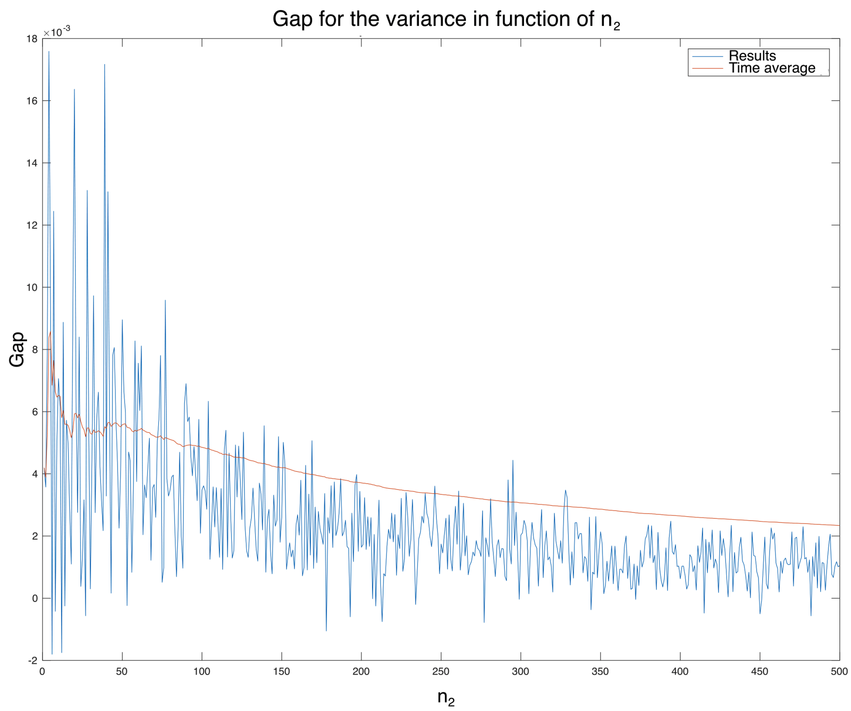


Figure 3.7: In blue, the average gap on 6 simulations between the true variance (see Notation) for a barbell graph made of $d = 15$ agents following n_2 and the true variance of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

In other words, a graph is vertex-transitive if its automorphism group acts transitively upon its vertices [MCZ15].

Definition 3.2.2. *An* ***automorphism*** *of a graph is a form of symmetry in which the graph is mapped onto itself while preserving the edge–vertex connectivity. Formally, an automorphism of a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a permutation σ of the vertex set $\mathcal{V}$, such that the pair of vertices (u, v) forms an edge if and only if the pair $(\sigma(u), \sigma(v))$ also forms an edge [Weia].*

3.2.1 Utility graph

The first sort of vertex-transitive graphs we will study are utility graphs, *i.e.*, the bipartite graph with partitions $\mathcal{V}_a$ and $\mathcal{V}_b$ where each agent of $\mathcal{V}_a$ is connected to all the agents of $\mathcal{V}_b$ and reciprocally and the size of partition $\mathcal{V}_a$ is equal to the size of partition $\mathcal{V}_b$. In Figure 3.8, we can see an example of such a graph with $n = 10$ agents.

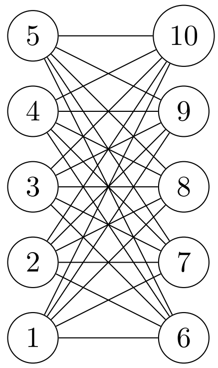


Figure 3.8: The utility graph with 10 nodes.

Now we will verify if the behaviour of this kind of graphs follows the one of the complete graph. To achieve this, we proceed in two steps: first of all we compute the fixed point of the variance of the complete graph using (2.5) and compare it with the numerical fixed point of the utility graph using (2.4). If we consider a complete graph and a utility graph; each one made of 50 agents and with a probability of replacement $p = 0.05$, we obtain for both graphs a fixed point for their variance equal to 0.0041. Secondly, we illustrate our results with a simulation. In Figure 3.9 we can see that the *variance of the utility graph converges to the one of the full-mesh graph.* Note that, these results do not contradict Figure 2.9. Indeed, we are considering a graph that is smaller than the one in Figure 2.9. Thus, its variance and the gap between its variance and the one of the complete graph are smaller. Moreover, the gap in this Figure was already pretty small.

3.2.2 n-cycle graph

The second sort of vertex-transitive graphs we will study are the n-cycle graphs. In Figure 3.10 is represented a 8-cycle graph. To begin, we try to identify how the steady state equation of this kind of graphs is different from the one of complete graphs. We proceed, as usual, by computing the expectation of the mean and of the mean square value. First of all, for the mean in a cycle, it is obvious that if we select an agent i for a gossip at time k, its value at time $k+1$ is either $\frac{x_i(k)+x_{i+1}(k)}{2}$ or $\frac{x_i(k)+x_{i-1}(k)}{2}$ with probability $\frac{1}{2}$ for both possibilities. Of course, $i+1$ and $i-1$ are here the agents that are adjacent to i. Given that, we can compute $\overline{x}(k+1)$:

$$\begin{aligned} n\mathbb{E}\left(\overline{x}(k+1)|x(k)\right) &= (n-2)\overline{x}(k) + 2\frac{1}{2}\mathbb{E}\left(\frac{x_i(k)+x_{i-1}(k)}{2} + \frac{x_i(k)+x_{i+1}(k)}{2}|x(k)\right) \\ &= n\overline{x}(k). \end{aligned}$$

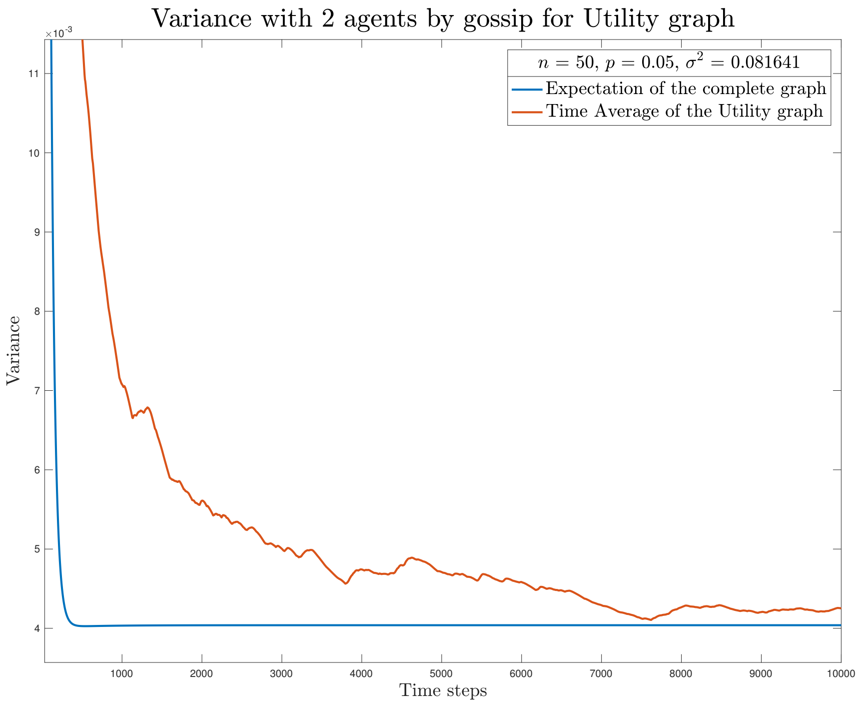


Figure 3.9: In blue, the expected variance for the complete graph using Equation (1.2). In red, the time-average (Definition 1.1.2) of the true variance (see Notation) of a utility graph of 50 agents with $p = 0.05$ over 10000 time steps.

Now we can apply the same reasoning as we did in the previous sections for the square mean value. As we will see, when we are working with graphs that differ from the complete graph, some simplifications do not hold anymore. For instance, if we consider the mean square value of this graph, we have:

$$n\mathbb{E}\left(\overline{x^2}(k+1)|x(k)\right) = (n-2)\overline{x^2}(k+1) + \frac{1}{4}\mathbb{E}\left(2x_i^2(k) + x_{i-1}^2 + x_{i+1}^2 + \boxed{2x_i(x_{i+1}+x_{i-1})}\right). \tag{3.1}$$

The framed part of the equation is the main issue of this computation. Indeed, for a non-complete graph, the statement $\mathbb{E}(x_i(k)x_{i+1}(k)|x(k)) = \mathbb{E}(x_i(k)x_{i-1}(k)|x(k)) = \overline{x}^2$ of the first chapter is no longer correct. Since we do not have any information about the value of i, $i+1$ and $i-1$ compared to the whole network, we do not know the value of the expectation of (3.1).

Now, let us consider a system made of 50 agents over 10000 time steps. The probability of replacement is $p = 0.05$ and, as previously, the value of each agent is uniformly distributed over $[-\frac{1}{2}, \frac{1}{2}]$. In Figure 3.11, we can see the gap between the expected variance of a complete graph and the one of the cycle graph defined above.

We conclude that *n-cycle graphs do not follow the same behaviour as full-mesh graphs*. Therefore, we can say that the steady state equation provided in Equation 1.2 does not hold for all the vertex-transitive graphs.

3.2.3 Degree of agents

As explained in Subsections 3.2.1 and 3.2.2, the vertex-transitivity property is not sufficient to follow the behaviour of the full-mesh graph. On the one hand, utility graphs fit quite well with the steady state equation of a complete graph where gossips are between two agents [HM17].

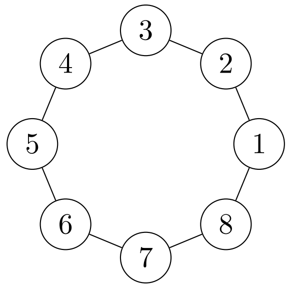


Figure 3.10: The cycle graph with 8 nodes.

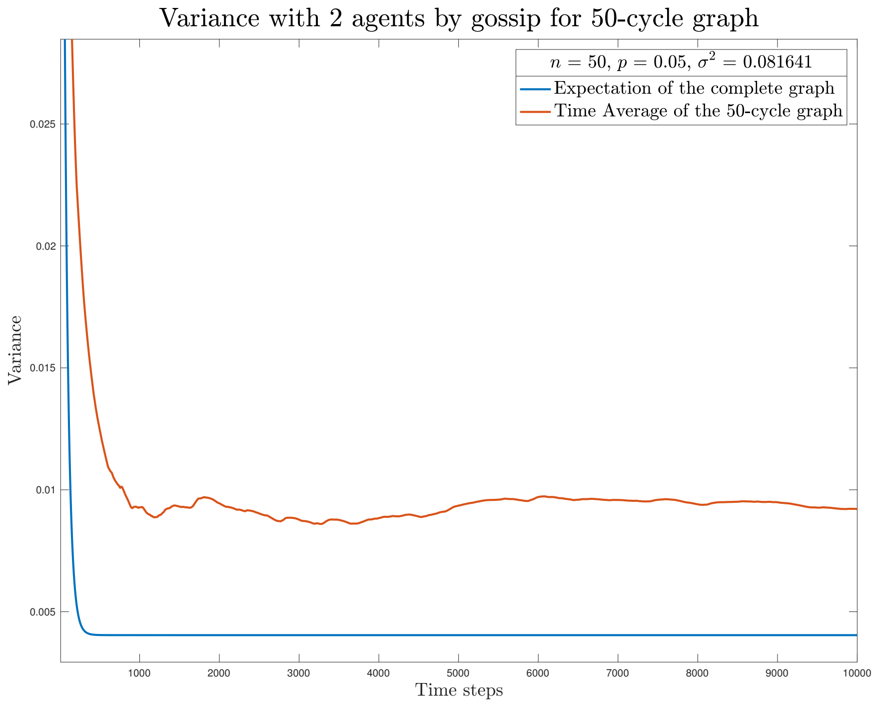
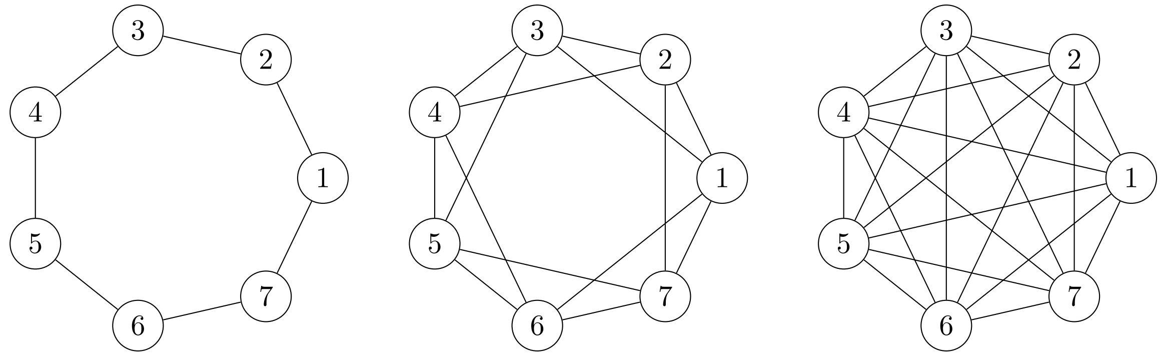


Figure 3.11: In blue, the expected variance for the complete graph using Equation (1.2). In red, the time-average (Definition 1.1.2) of the true variance (see Notation) of a 50-cycle graph with $p = 0.05$ over 10000 time steps.

On the other hand, n-cycle graphs provide a gap between the expected variance for a complete graph and the obtained results.

In this section, we will focus on an additional characteristic of our network: its diameter (Definition 3.3.1). In such a way, we will be working on vertex-transitive graphs whose diameter becomes smaller and smaller after each step. The first graph will be the n-cycle graph whilst the last one will be the full-mesh graph with the same number of agents. For example, in Figure 3.12, we can see a system made of seven agents. The degree of each agent increases by two at each update until the full-mesh structure is encountered. Note that, following this evolution, for an initial n-cycle graph where n is odd, we will have $\frac{n-1}{2}$ steps until the diameter is equal to one. Moreover, if the diameter at step k was $d_{\mathcal{G}}$, at step $k+1$ it will be $\lceil \frac{d_{\mathcal{G}}}{2} \rceil$.

Now, let us have a look at the evolution of the variance of the considered vertex-transitive graphs. We will also have a look at the gap between their variance at the last time steps, *i.e.*, $t = 50000$ and the variance obtained with a complete graph with the same number of agents at



(a) $d_{\mathcal{G}} = 3$ (b) $d_{\mathcal{G}} = 2$ (c) $d_{\mathcal{G}} = 1$

Figure 3.12: Vertex-transitive graphs with different diameters

the same time $t = 50000$. In Figure 3.13, different variances are shown as a function of the degree of each agent in the system. We can see that the distance from the upper one to the second upper's is the greatest one. This distance quickly decreases as it is confirmed in Figure 3.14. This behaviour is quite positive. Indeed, a linear decrease would have meant that the steady state equation of the complete graph is not a good approximation for others. The hyperbolic slope encountered, at the contrary, means that after a certain degree, *i.e.*, a given value of our diameter, the system will behave almost as a complete graph.

This conclusion is very interesting since it constitutes a first explanation for the behaviour of the utility graph compared to the one of the n-cycle graph. We have now the intuition that the more a graph will be connected, the more its behaviour will be close from the one of the complete graph, *i.e.*, the more its variance will be small. This observation weights in favour of Conjecture 3.0.1 but does not constitute a proof. Indeed, it shows that the Conjecture holds for the considered vertex-transitive graphs. In the next section, we will try to verify whether the vertex-transitivity is an important property or if it is possible to look only at the diameter of a graph in order to guess the behaviour of its variance.

3.3 Role of the Graph Diameter

Section 3.2 was focused on vertex-transitivity. We have seen that for such graphs, the more the graph is connected and thus the more the diameter is small, the more the variance of the considered system tends towards the one of the full-mesh graph. Now, we will try to generalize this conclusion, *i.e.*, to verify what happens when working on a graph without any particular structure, but whose diameter will decrease until it is equal to one, *i.e.*, until the graph is complete.

Definition 3.3.1. *The* ***diameter*** *of a graph $\mathcal{G}$ $d_{\mathcal{G}}$ is equal to the length $max_{(u,v)}d(u,v)$ of the "longest shortest path" between any two graph vertices $u, v \in \mathcal{V}$ of a graph, where $d(u,v)$ is a graph distance [Weib].*

Through this section, we will mainly be working considering random graphs (Definition 3.3.2).

Definition 3.3.2. ***Random graph*** *is the general term to refer the probability distribution over graphs. Random graphs may be described simply by a probability distribution, or by a random process that generates them. For instance, a random graph can be obtained by starting with a set of v isolated vertices and adding successive edges between them at random [FK15]. Through this document, we will consider two random graph models:*

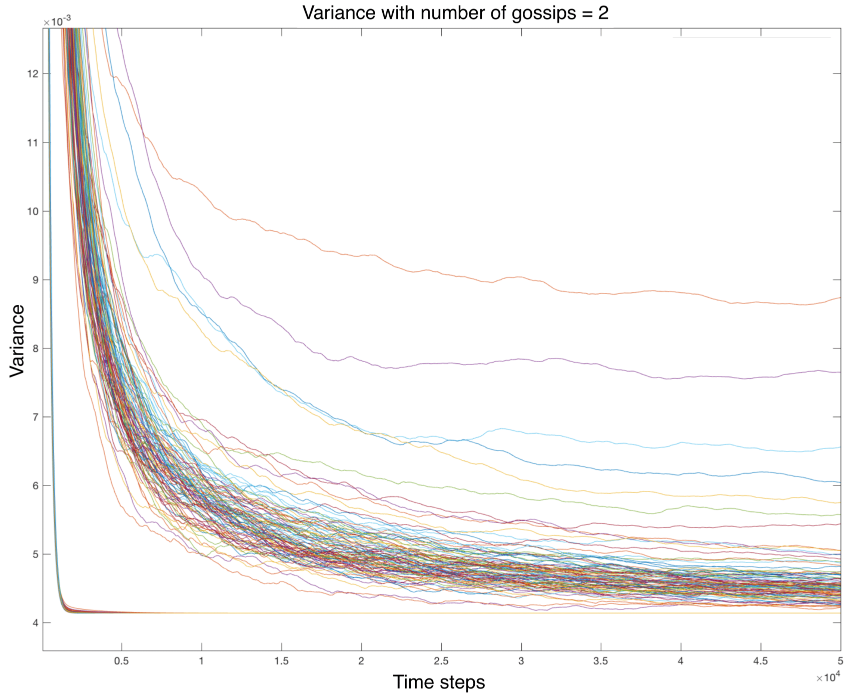


Figure 3.13: Variance (see Notation) of vertex-transitive graphs following the degree of each node compared to the expected variance of the complete graph with 201 agents and $p = 0.05$ using Equation (1.2). The curve at the bottom of the Figure is the expected variance of the complete graph.

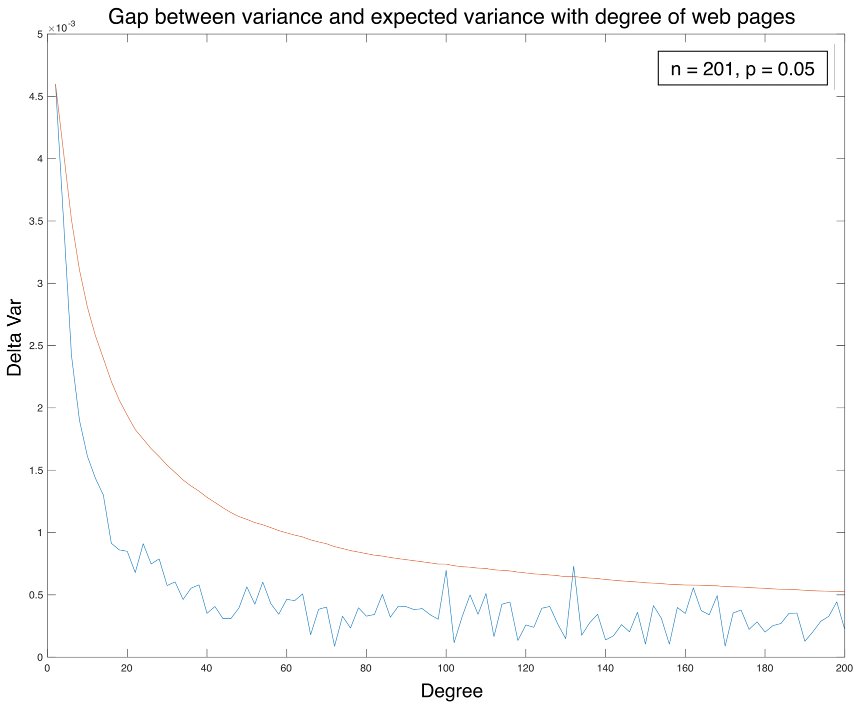


Figure 3.14: Gap between the true variance (see Notation) of vertex-transitive graph following the degree of each agent in the graph and the true variance of the complete graph with the same number of agents, $n = 201$, and $p = 0.05$.

1. *Erdos-Renyi model: two agents randomly selected are connected with probability $P_c \in [0, 1]$. When $P_c = 1$, the graph is no longer random and is a complete graph.*

2. *Configuration model: if we consider a system made of n agents, a degree sequence of length n will be provided such that the degree of the agents is known.*

To further investigate about the link between variance of our system and its diameter, let us consider a network made of 100 agents following Erdos-Renyi model (Definition 3.3.2) and probability of replacement $p = 0.05$. In this context, we will try different values for P_c in order to obtain different values for the diameter of the considered graphs. In Figure 3.15, which is obtained with 100 simulations for each 200 different values of P_c, an interesting behaviour can be observed: *the growth of the variance is not linear.* Indeed, we can observe that it seems to have an asymptotic value. Figure 3.16, that has been obtained with 100 simulations, represents the average value of the variance for a graph made of 100 agents with a diameter equal to 99. This figure confirms the statement about the increase of the variance.

In Figure 3.17, we can see the evolution of the variance of a barbell graph made of 200 agents whose diameter grows by 2 at each step. Obviously, each time the diameter is incremented by two, a node is removed from both complete graphs at the extremities of the barbell graph. In this Figure, we can see that the variance does not behave monotonically. First of all, it increases for graphs with small diameter. Then, it decreases when the diameter becomes larger. For that reason, it becomes very difficult to formulate a general rule from the diameter of a graph.

In all the considered cases, we saw that when the diameter is small, the variance increases following the growth of the diameter. On the other hand, for large diameters, it is no more possible to conclude anything. These results motivate us to find another element in order to strengthen Conjecture 3.0.1.

3.4 Eigenvalues of the Expected Transition Matrix of the System

Until now, we tried to find a correlation between the variance of a vertex-transitive graph and the variance of the full-mesh graph. We also studied the behaviour of the variance of a graph following its diameter. These two methods have presented some limits in their utility and above all, they do not constitute a proof. We will thus be focused on another element: the eigenvalues of the expected transition matrix (Equation (3.2)) for the steady state equation associated to the graph we are working on. Indeed, in [BGPS06], the authors suggest that it is possible to link the rate of convergence and the second highest eigenvalue of the expected transition matrix. The main idea is explained below: let us define the rate $y(t) = x(t) - x_\infty$ at which the error converges to 0, where $x_\infty = \lim_{t\to\infty} x(t) = x_{\text{ave}}\mathbf{1}$. We have also to define a transition matrix, $W(t)$, when a gossip occurs, *i.e.*, $x(t+1) = W(t)x(t)$ and if the gossip is between agents i and j at time t:

$$W(t) = W_{ij} = I - \frac{(e_i - e_j)(e_i - e_j)^\top}{2}.$$

The iteration of transition matrix intends to compute the average, and therefore must preserve sums: this means that $\mathbf{1}^\top W(t) = \mathbf{1}^\top$, where $\mathbf{1}$ denotes the vector of all ones. Also, the vector of averages must be a fixed point of the iteration, *i.e.*, $W(t)\mathbf{1} = \mathbf{1}$.

Definition of gossips depends on the author. It is not the same in the articles [BGPS06] and [RFTI15]. Note that in this context, the way gossips are defined is very important. Previously, we were randomly selecting an edge in $\mathcal{E}$. This method therefore gave a bigger probability to agents with an important degree of being implied in a gossip. In [BGPS06], gossips are defined as follows: an agent is selected among the n agents of the system randomly with the same

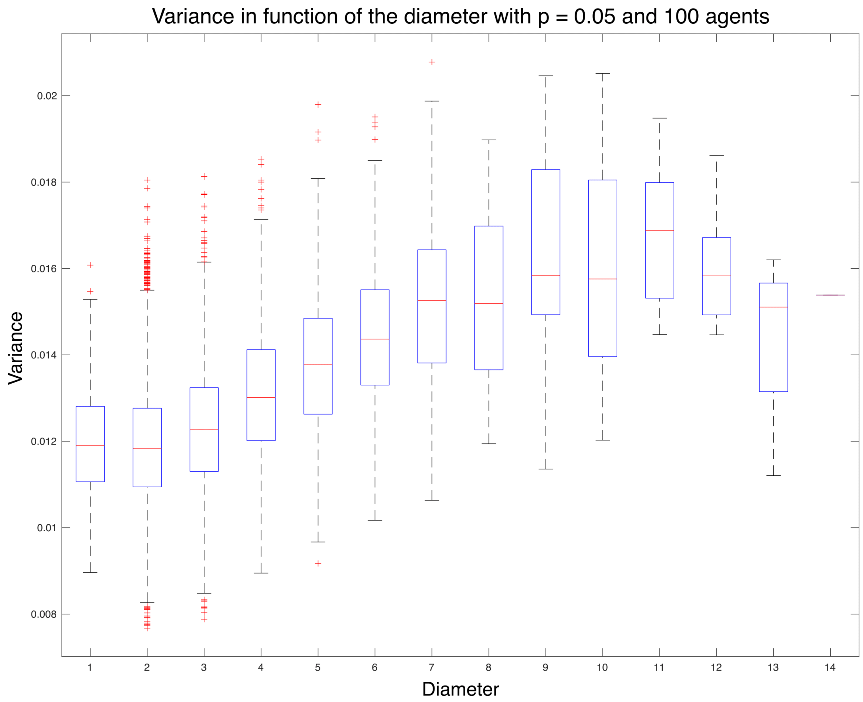


Figure 3.15: Average variance of the considered random graphs following Erdos-Renyi (Definition 3.3.2) model as a function of their diameter. $n = 100$, $p = 0.05$.

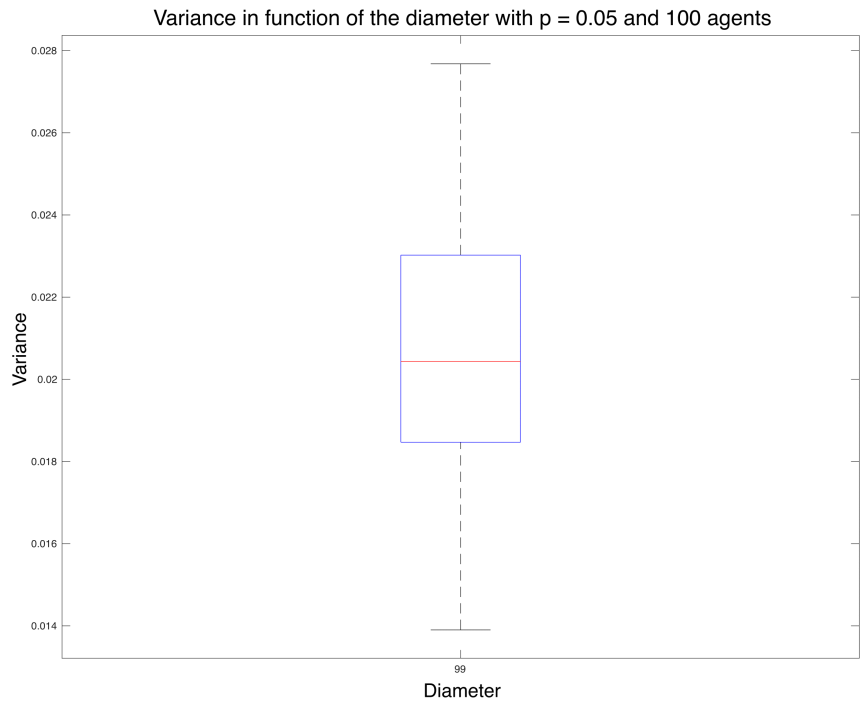


Figure 3.16: Average variance of the random graphs following Erdos-Renyi (Definition 3.3.2)k with a diameter equal to 99. $n = 100$, $p = 0.05$.

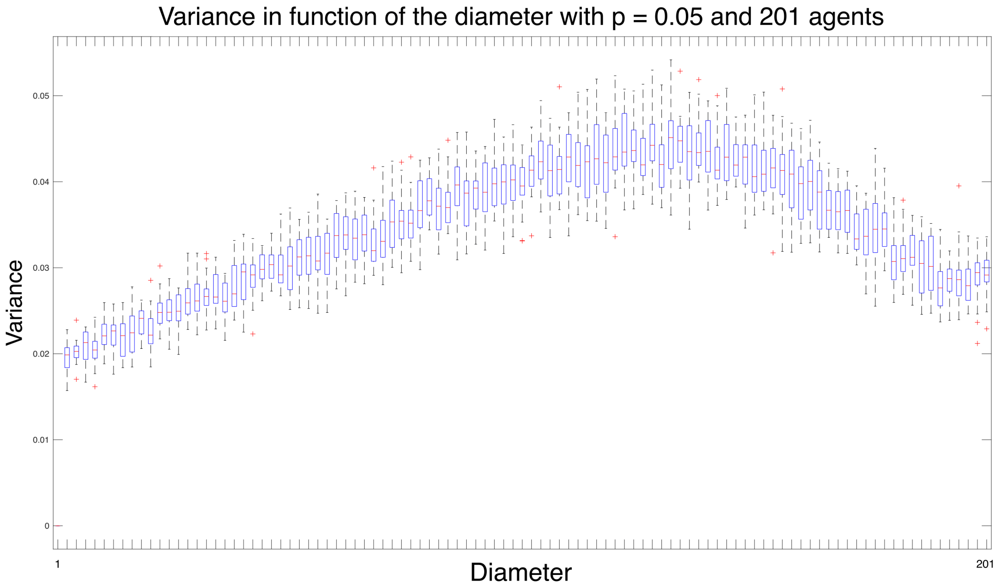


Figure 3.17: Average variance of barbell graphs (Definition 3.1.1) as a function of their diameter. $n = 200$, $p = 0.05$. The number of agents stays the same, but we modify n_2 and d such that the diameter increases.

probability for each agent. Then we look at the agents that are connected to the one we have chosen, and we take one randomly with the same probability for each adjacent agent. This definition is different from the previous one since agents with an important degree are less advantaged as they were before. In the definition where all the edges have the same probability of being implied in a gossip, the upper bound obtained in [BGPS06] does not hold. Indeed, after a few[1] iterations, the time-average (Definition 1.1.2) of the variance is above this upper bound, whatever the time-window (Definition 1.1.3) selected for the computation of the time average.

Given this, we consider the evolution of y:

$$\begin{aligned} y(t+1) &= x(t+1) - x_{\text{ave}}\mathbf{1} \\ &= W(t)x(t) - x_{\text{ave}}W(t)\mathbf{1} \\ &= W(t)\left(x(t) - x_{\text{ave}}\mathbf{1}\right) \\ &= W(t)y(t). \end{aligned}$$

Thus, the variance of our system can be written as:

$$\begin{aligned} \operatorname{Var}\left(x(t+1)\right) &= \frac{1}{n}\mathbb{E}\left(y(t+1)^\top y(t+1)\right) \\ &= \frac{1}{n}\mathbb{E}\left(y(t)^\top W(t)^\top W(t) y(t)\right). \end{aligned}$$

Moreover, note that:

$$\begin{aligned} W(t)^\top W(t) &= \left(I - \frac{(e_i - e_j)(e_i - e_j)^\top}{2}\right)^2 \\ &= I - \frac{(e_i - e_j)(e_i - e_j)^\top}{2} \\ &= W(t). \end{aligned}$$

[1] a lot of iterations (about 700000)

This result is not surprising since $W(t)$ is actually a projection. Now, we will see the difference between gossips such as defined in [BGPS06] and gossips where we select an edge with probability $\frac{1}{|\mathcal{E}|}$. With the definition of [BGPS06], we have an expected transition matrix W equal to:

$$\mathbb{E}(W(t)) = W \triangleq \frac{1}{n} \sum_{i,j} P_{ij} W_{ij},$$

where P_{ij} is the probability that if agent i is selected for a gossip, node j is also selected. Indeed, each transition matrix W_{ij} will be selected with a probability $\frac{P_{ij}}{n}$. With our definition, W will be slightly different:

$$\mathbb{E}(W(t)) = W \triangleq \frac{1}{|\mathcal{E}|} \sum_{i,j} M_{ij} W_{ij}, \tag{3.2}$$

where M_{ij} is the adjacency matrix and is defined as follows:

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

From now on, we will use the definition of the expected transition matrix from Equation (3.2).

Given this, it is possible to find a lower and an upper bound for the evolution of our variance:

$$\begin{aligned} \mathrm{Var}(x(t+1)) &= \frac{1}{n} \mathbb{E}\left(y(t)^\top W(t)^\top W(t) y(t) \right) \\ &= \frac{1}{n} y(t)^\top \mathbb{E}\left(W(t)\right) y(t) \\ &= \frac{1}{n} (x(t) - x_{\mathrm{ave}} \mathbf{1})^\top \mathbb{E}\left(W(t)\right) (x(t) - x_{\mathrm{ave}} \mathbf{1}). \end{aligned}$$

Since in our case, the values of $x(\cdot)$ are uniformly distributed on $[-\frac{1}{2}, \frac{1}{2}]$ with mean 0, we can say that:

$$\mathrm{Var}(x(t+1)) = \frac{1}{n} x(t)^\top \mathbb{E}\left(W(t)\right) x(t).$$

Thus, if $1, \lambda_2, \ldots, \lambda_n$ are the eigenvalues of W such that $1 \geq \lambda_i \geq \lambda_{i+1} \quad \forall i = 2, \ldots, n-1$, we can say knowing that $y \perp \mathbf{1}$ [BGPS06]:

$$\lambda_n(W) \frac{\|x(t)\|^2}{n} \leq \mathrm{Var}(x(t+1)) \leq \lambda_2(W) \frac{\|x(t)\|^2}{n},$$

$$\lambda_n^{t+1}(W) \frac{\|x(0)\|^2}{n} \leq \mathrm{Var}(x(t+1)) \leq \lambda_2^{t+1}(W) \frac{\|x(0)\|^2}{n}. \tag{3.3}$$

3.4.1 Bounds on the variance without replacement

First of all, we work on simulations where no replacement occurs. In this case, Equation (3.3) can be used without any additional precaution. We are interested in two things, first of all the quality of the upper and lower bound compared to the time-average on the variance we obtain. Second of all, we will try to establish a relation between the eigenvalues of a system and the evolution of its variance.

In Figure 3.18, we can see lower bounds and upper bounds for the complete graph (CG), the Barbell graph (BG), the cycle graph (Cyg) and the random graph (RG) following Erdos-Renyi model (Definition 3.3.2) with $P_c = 0.8$. For all the cases, the number of agents is equal to 100. We can see that the upper and lower bound of the complete graph are the same. Note also that the upper bounds of the complete graph is the smallest upper bound in this Figure. Now, we want to go one step further by adding replacements in the computation of our bounds.

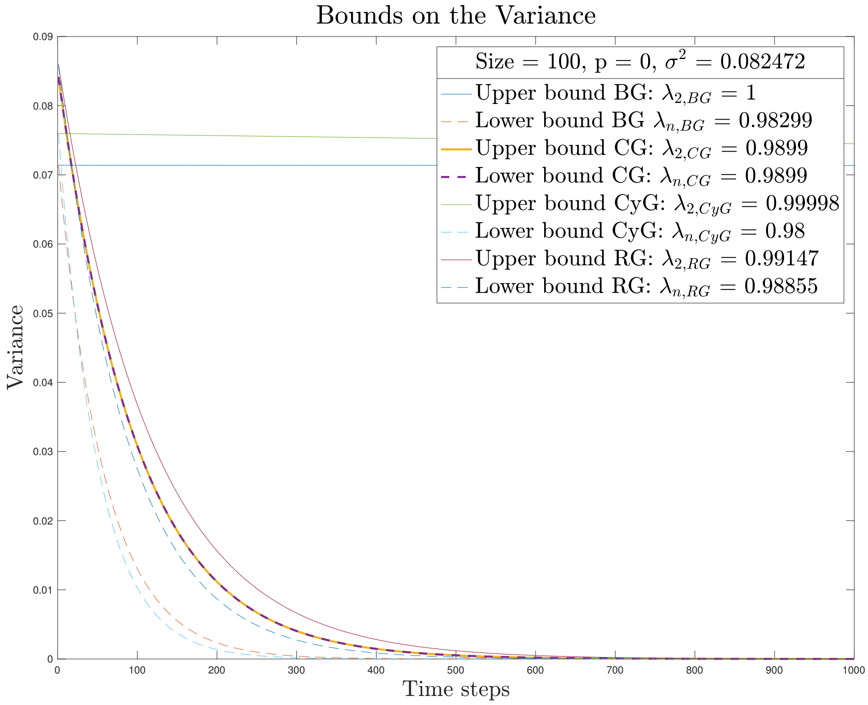


Figure 3.18: Bounds on the true variance (see Notation) using Equation (3.3) following the considered graph. When the curve is dotted, then it represents a lower bound. Otherwise, it is an upper bound.

3.4.2 Bounds on the variance with replacements

Equation (3.3) tells us how the variance is bounded from a time step to the next one when only gossips occur. It would be interesting to wonder, *how this variance behaves when we also include replacements in our system.* To achieve this, we use steady state equation (1.6) of Subsection 1.1.2 which says that for a replacement:

$$\mathbb{E}X(k+1) = A_r \mathbb{E}X(k) + b_r,$$

where

$$A_r = \begin{pmatrix} \frac{n-2}{n} & \frac{1}{n^2} \\ 0 & \frac{n-1}{n} \end{pmatrix} \quad \text{and} \quad b_r = \begin{pmatrix} \frac{\sigma^2}{n^2} \\ \frac{\sigma^2}{n} \end{pmatrix}.$$

Thus, the expected variance when a replacement occurs evolves following:

$$\begin{aligned} \mathbb{E}\text{Var}(x(k+1)) &= \left(\frac{n-1}{n}\mathbb{E}\overline{x^2}(k) + \frac{\sigma^2}{n}\right) - \left(\frac{n-2}{n}\mathbb{E}\overline{x}^2 + \frac{1}{n^2}\mathbb{E}\overline{x^2} + \frac{\sigma^2}{n^2}\right) \\ &= \frac{n-2}{n}\mathbb{E}\text{Var}(x(k)) + \frac{1}{n}\mathbb{E}\overline{x^2}(k) - \frac{1}{n^2}\mathbb{E}\overline{x^2}(k) + \frac{1}{n}\left(1 - \frac{1}{n}\right)\sigma^2. \end{aligned}$$

Since $\mathbb{E}\left(x^2(k+1)\right) = \sigma^2$, we obtain:

$$\mathbb{E}\text{Var}(x(k+1)) = \frac{n-2}{n}\mathbb{E}\text{Var}(x(k)) + \frac{2}{n}\left(1 - \frac{1}{n}\right)\sigma^2.$$

Therefore, if a replacement occurs with probability p and a gossip occurs with probability $(1-p)$, then we have using the bounds provided in Equation (3.3) on which we take the expectation:

$$\mathbb{E}\text{Var}(x(t+1)) \geq (1-p)\lambda_n(W)\mathbb{E}\text{Var}(x(k)) + p\left(\frac{n-2}{n}\mathbb{E}\text{Var}(x(k)) + \frac{2}{n}\left(1-\frac{1}{n}\right)\sigma^2\right) \tag{3.4}$$

$$\geq \left((1-p)\lambda_n(W) + p\frac{n-2}{n}\right)\mathbb{E}\text{Var}(x(k)) + \frac{2}{n}p\left(1-\frac{1}{n}\right)\sigma^2, \tag{3.5}$$

and

$$\mathbb{E}\text{Var}(x(t+1)) \leq (1-p)\lambda_2(W)\mathbb{E}\text{Var}(x(k)) + p\left(\frac{n-2}{n}\mathbb{E}\text{Var}(x(k)) + \frac{2}{n}\left(1-\frac{1}{n}\right)\sigma^2\right) \tag{3.6}$$

$$\leq \left((1-p)\lambda_2(W) + p\frac{n-2}{n}\right)\mathbb{E}\text{Var}(x(k)) + \frac{2}{n}p\left(1-\frac{1}{n}\right)\sigma^2. \tag{3.7}$$

We now try these bounds for different graph. In Figure 3.19, we can see the variance of a system, which is made of 100 agents and with replacement probability $p = 0.05$, in red and the obtained upper bound above in blue. We can see that for the graphs we have tested, the upper bound behaves as it should, *i.e.*, it is always above the variance in red. However, for the complete graph we can see that sometimes the variance is bigger than the upper bound. This phenomenon is logical since for the complete graph the lower and the upper bounds are the same (see Figure 3.18). However, for the replacements, we consider the average case. Thus, is it possible that the variance comes above this average value such that it also exceeds the upper bound.

To be more general, if the graph we are considering is quite similar to the complete graph or if its variance is more or less the same as the one of the complete graph, then it is possible that its variance exceeds its upper bound since for the replacement we have used the average case.

Note that in Figure 3.19, we have not drawn the lower bound since we have seen in Figure 3.18 that the slower the upper bound decreases, the faster the associated lower bound diminishes.

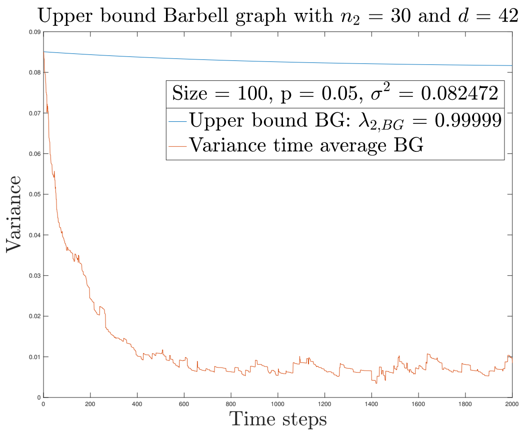
We can thus see that the upper bound found is above our results as it was expected, except for the complete graph, as we discussed earlier. These results end this part. We have not proven why non complete graphs have a greater variance than complete graph with the same number of agents. However, now we have intuition about the behaviour of the variance of a graph given its upper bound. The slower this upper bound will decrease, the most its variance will be above the one of a complete graph with the same number of nodes.

Nonetheless, we proved that the upper bound for the complete graph U_C is always below the upper bound for any other graph U_{NC}. Indeed, we know that the more the graph will be connected, the smaller the second eigenvalue of its expected transition matrix will be. Moreover, we observe that the behaviour of a graph is nearest the one of the upper bound than the lower bound.

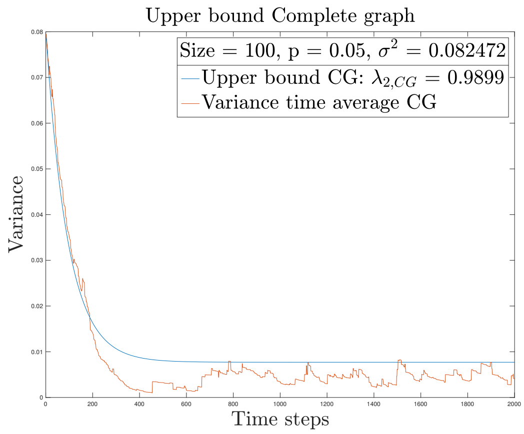
3.5 Conclusion

In this chapter we tried to prove that the variance of the complete graph can be used as a lower bound for the variance of any graph with the same number of agents. First of all, we verified if the steady state equation for the complete graphs was only for this sort of graphs or also for a broader type of graphs such as vertex-transitive graphs. If utility graphs behaved very closely to the complete graphs, the $n-$cycle graph confirmed that the vertex-transitivity property was not enough for using the steady state equation of the complete graph.

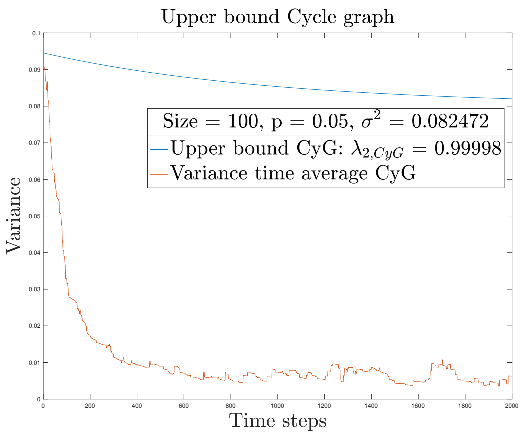
Secondly, we tried to find a link between the diameter of a graph and its variance. To achieve this, we computed the variance of some graphs with different diameters after a given number of time steps. For graphs with a small diameter and the same number of agents, we observed that



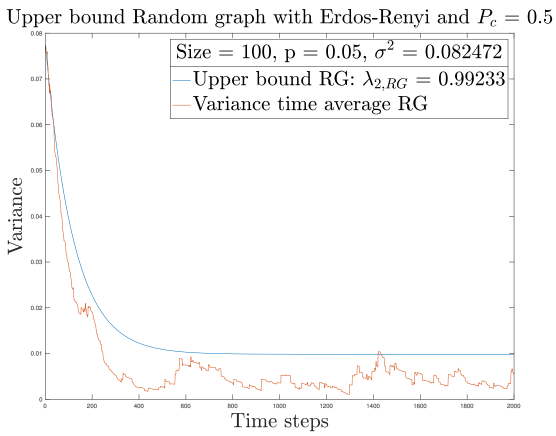
(a) Barbell graph where $n = 30$, $d = 42$ and $p = 0.05$



(b) Complete graph with $n = 100$ and $p = 0.05$



(c) Cycle graph with $n = 100$ and $p = 0.05$



(d) Random graph with $n = 100$, $P_c = 0.5$, $p = 0.05$

Figure 3.19: True variance (see Notation) and its upper bound using (3.7) for a given graph typology

a growth in the diameter ensues a growth in the variance of the graph. Unfortunately, when we work with graphs with large diameter, this conclusion is no longer true.

Since the researches about the diameter of a graph were not conclusive, we tried another approach for proving Conjecture 3.0.1: we looked at the eigenvalues of the expected transition matrix of the graph for finding upper and lower bounds for the evolution of the variance. This method provided results that are quite interesting since we proved that the upper bound of the complete graph is always smaller than the upper bound for any other graph. Moreover, we saw that a system is more influenced by its second largest eigenvalue, *i.e.*, by its upper bound than by its lower bound. For this reason, since the upper bound of the complete graph is below the one of any other graph, we have good reason to believe that the variance of the complete graph can be seen as a lower bound for the variance of any other graph.

Chapter 4

Randomized PageRank Computation in Open Systems

In the previous chapters the system could, for each event, either replace the value of an agent or make a gossip between two or more agents. All the modifications were then depending on those two kinds of events. Now we append a new degree of freedom by adding a random vector whose size is equal to the number of agents. At each event/time step, each agent receives a compensation from this random vector, and we analyze how the system behaves after a sufficiently large number of events. We are thus considering the case of systems with input, since at each step we add something to the value held by each agent. Another difference is that we consider directed graphs while we were working with undirected graphs until now. Finally, from now on, replacements modify not only the value of an agent, but also the structure of the considered system. Indeed, each time a replacement occurs, the leaving agent deletes all its adjacent links while the entering agent creates links towards other agents following a given property. The other agents point this new agent with the same property.

We devote this chapter to a well-known application: we consider the PageRank algorithm which is used to determine the popularity of a given web page. For this, the input vector plays an important role in the evolution of the importance of a page: it avoids to fall in a dead-end by adding a restart probability as we will see later. First of all, we present a classical algorithm for the computation of the popularity of the web pages. We will see that this method has some limitations due to the computations it requires. Then, we present an alternative to this Classical PageRank that uses open multi-agent systems. The first version of this algorithm comes from [RFTI15] and can cope with system where only gossips occur, *i.e.*, with multi-agent systems. For this reason, we extend this method such that it is able to support replacements, and we study the strengths and weaknesses of this new algorithm. We will thus try to adapt it such that it works in an open multi-agent systems environment. Finally, we compare the results obtained with the Classical PageRank and the PageRang using open multi-agent systems, *i.e.*, Randomized PageRank in order to discuss their accuracy.

4.1 Classical PageRank

The Classical PageRank is a method for computing the popularity of web pages. The more important the PageRank of a page, the more popular and important it is.

4.1.1 Algorithm

This section is very inspired form [SB98] and [JL14]. Consider the Web as a directed graph where nodes are web pages and there is an arc from i to j if there is one or more references from i to j. For this algorithm, we define a new concept : the transition matrix (Definition 4.1.1).

Table 4.1: PageRank of the graph in Figure 4.1.

Iteration	A	B	C	D
1	0.25	0.25	0.25	0.25
2	0.3625	0.2125	0.2125	0.2125
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	0.3276	0.2241	0.2241	0.2241

Definition 4.1.1. *In general, we use the* ***transition matrix*** *of the Web to describe what happens to random surfers after one step. This matrix M has n rows and columns, if there are n pages. The element m_{ij} in row i and column j has value $1/d_{out}$ if page j has d_{out} arcs out, and one of them is to page i. Otherwise, $m_{ij} = 0$.*

In Figure 4.1, we can see a graph made of four web pages. The transition matrix associated to this graph is:

$$M = \begin{pmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{pmatrix}.$$

Classical PageRank works as follows:

1. Give an initial PageRank's vector $v_i(0) = \frac{1}{n} \quad \forall i = 1, \ldots, n$
2. Update the PageRank of each web page following:

$$v(t+1) = \beta M v(t) + (1-\beta)\frac{e}{n}, \tag{4.1}$$

where $\beta \in [0,1]$ is the *restart probability*. This parameter is added in order to avoid that the algorithm gives a PageRank equal to 1 for a web page with no out-link but with in-links and $\frac{e}{n}$ is the input vector where e is a vector with all its entries equal to 1 and n is the number of agents of the graph. It means that at each step, we distribute a small amount of the popularities to each page. Note that the solution of this Classical PageRank is the fixed point of the Equation (4.1). For the graph in Figure 4.1, the PageRank of the four web pages taking $\beta = 0.1$ is given in Table 4.1. The fixed point is provided for the iteration n. In addition, if our system is made of n pages, we add the following constraint:

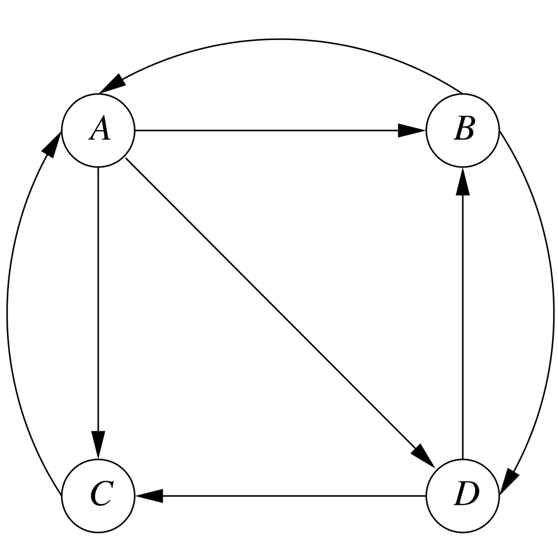


Figure 4.1: Web made of four web pages [JL14]

$$\sum_{i=1}^{n} x_i(k) = 1 \quad \forall \; k \geq 0. \tag{4.2}$$

This constraint expresses the fact that the sum of all the popularities is a bounded value. Moreover, the vector containing the PageRank of all our web pages is normalized by definition of Equations (4.1) and (4.2).

4.1.2 Discussion

This method has the advantage that it is very simple to implement and very intuitive. However, it has some issues. First of all, it requires many computations for each update of the PageRank. When working with large systems, the amount of computations becomes difficult to deal with. Secondly, this algorithm requires to know the structure of the graph. This last point is a very important one since the web contains *at least 4.46 billion pages (Tuesday, 29 May, 2018)* [dK]. In this context, knowing all the connections from a page to another one is a huge information. Moreover, if the graph has a structure that is modified with time, the Classical PageRank will not be able to adapt its computation. It means that the graphs we are considering must have a frozen structure. Of course, in practice, this hypothesis is a very strong constraint. For these reasons, tools have been developed and in the next pages we will try to explore and extend one of those thanks to [RFTI15].

4.2 Randomized PageRank with Open Systems

The Classical PageRank works pretty well for small graphs whose structure is known and not modified. However, when the number of web pages of the system becomes too large or when some directed edges are removed and/or added, the algorithm begins to present some lacks. Indeed, the Classical PageRank, as it is defined in Section 4.1, can not cope with modifications in the structure of the graph and implies a number of computations difficult to manage.

There exist a lot of alternatives for the Classical PageRank. Some of them are explained in Chapter 5 of [JL14]. Through this section, we will explore a method using open multi-agent systems. This method will return the popularity of the web pages of the considered network. From now on, an agent refers to a web page whilst an edge is a reference from a page to another. Note that using open multi-agent systems is very interesting for two reasons. The first one because we use a decentralized method. Thus, we no longer need to know the structure of the network. Each agent knows its value and its connections with other agents, but we no longer have a centralized entity aware of the structure of the graph. Secondly, using *open* systems allows us to consider a network with a structure that is changing. This last point is very important since the content and the references of a web page are not something frozen.

First of all it is important to answer to the question: *How to link PageRank,* i.e.*, an algorithm using systems with input and open multi-agent systems in the sens we defined it previously?* We can create a parallelism between both since web pages can be considered as agents whose value would be the popularity of the page. A page can be created and deleted. Thus, it is possible to imagine a replacement, and we will explain in the next sections how they are defined. Directed edges would be a reference from an agent, *i.e.*, a web page, to another one. The last point who needs to be elucidated is the way gossips are defined.

Of course, the main advantage of Randomized PageRank with open systems is its capacity to support modifications in the structure of the graph. For this reason, we will explain separately how the algorithm works when replacements are possible or not.

4.2.1 Gossip

In [RFTI15], the authors suggest the following possibility for the gossips: at each step, we select randomly a directed edge adjacent to two agents i and j, *i.e.*, two web pages, from i to j, and we update their value from step k to step $k+1$ according to:

$$x_i(k+1) = (1-r)\left(1 - \frac{1}{d_{out,i}(k)}\right) x_i(k) + \frac{r}{n}, \tag{4.3}$$

$$x_j(k+1) = (1-r)\left(x_j(k) + \frac{1}{d_{out,i}(k)} x_i(k)\right) + \frac{r}{n}, \tag{4.4}$$

$$x_h(k+1) = (1-r)x_h(k) + \frac{r}{n} \quad \text{if} \quad h \neq i, j. \tag{4.5}$$

Where $d_{out,i}(k)$ is the number of out-links of page i at time k and $\frac{r}{n}$ for each equation expresses the input value. An edge is then selected with a probability $\frac{1}{|\mathcal{E}|}$. The value of r is defined as it is in [RFTI15], *i.e.*,

$$r = \frac{\beta}{(\beta - \beta|\mathcal{E}| + |\mathcal{E}|)},$$

where β is the restart probability associated to the PageRank algorithm. Note that, for this algorithm, Equation (4.2) still holds and the vector containing all the popularities stay normalized by definition of Equations (4.3), (4.4), (4.5) and (4.2). A very important point is that Equations (4.3), (4.4) and (4.5) suggest that we have to know the global structure of the network. However, we have seen previously that the main advantage of the Randomized PageRank is that it is a decentralized method. To cope with this point the reader is invited to consider Remark 4 of [RFTI15]. This Remark says that it is possible to stay in a decentralized context by introducing clocks on each link of the network. This point will no longer be discussed in this document but it is important to precise that this definition does not compromise the decentralized property. Note that the idea is similar to the use of Poisson clock (Definition 1.0.3) in Chapter 1. Equations (4.3), (4.4) and (4.5) have not been defined by chance. A proof is given in [RFTI15] and demonstrates that the time-average (Definition 1.1.2) of the value held by an agent converges towards the fixed-point (Definition 1.1.1) of Equation (4.1) when $t \to \infty$. We can also wonder how to interpret these three equations. The edge randomly selected comes from i and is pointing towards j. Equations (4.3) and (4.4) express the fact that agent i gives a part of its popularity to agent j while Equation (4.5) represents the behaviour of all the other agents. We have thus presented a decentralized method of the PageRank computation.

Now, we will go outside the framework of [RFTI15] and deepen some results that come from this article. First of all, we will be focused on the average behaviour of this sort of network with time. To achieve that, we first compute the steady state equation (Definition 1.0.4) when the only event that occurs at each time step is a gossip between two pages. As describe in Equations (4.3), (4.4) and (4.5), the value of each agent will change. To begin, let us define $X(k) = (\overline{x} \;\; \overline{x}^2 \;\; \overline{x^2})^T$.

Hypothesis 4.2.1. *We assume that whatever the structure of the graph:*

$$\mathbb{E}\left(x_i(k)xj(k) - \overline{x}^2(k)\right) = 0.$$

The assumption is true in the case of the complete graph. For other graphs the equation does not hold. However, for computation simplicity, referring to this hypothesis means that the equality holds.

Proposition 4.2.1. *The steady state equation (Definition 1.0.4) of the Randomized PageRank under Hypothesis 4.2.1 is given by the following:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} (1-r) & 0 & 0 \\ \frac{2r}{n}(1-r) & (1-r)^2 & 0 \\ \frac{2r}{n}(1-r) & \frac{2}{nd} & \frac{(1-r)^2}{n}\left(n + \frac{2}{D} - \frac{2}{d}\right) \end{pmatrix} \mathbb{E}X(k) + \begin{pmatrix} \frac{r}{n} \\ \frac{r^2}{n^2} \\ \frac{r^2}{n^2} \end{pmatrix}. \tag{4.6}$$

where $d = \mathbb{E}(d_{out,i}(k))$ is the average out-degree of a node and $D = \mathbb{E}(d_{out,i}^2(k)) = \mathbb{E}(d_{out,i}(k))^2 + \mathrm{Var}(d_{out,i}(k))$ is the average of the square out-degree of a node.

Proof. We first verify the results obtained for the square mean value for a system of n pages. We know that from step k to step $k+1$, if the two selected agents are (i,j) and i points towards j, our system becomes:

$$\begin{aligned}
n\mathbb{E}\left(\overline{x}(k+1)|x(k)\right) &= \mathbb{E}\left(\sum_{h=1,h\neq i,j}^{n}\left[(1-r)x_h(k)+\frac{r}{n}\right]\right.\\
&\left.+(1-r)\left(\left(1-\frac{1}{d_{out,i}(k)}\right)x_i(k)+x_j(k)+\frac{1}{d_{out,i}(k)}x_i(k)\right)+2\frac{r}{n}\right)\\
&= (n-2)\left[(1-r)\overline{x}(k)+\frac{r}{n}\right]+2(1-r)\overline{x}(k)+2\frac{r}{n}\\
&= n(1-r)\overline{x}(k)+r.
\end{aligned}$$

With the following results, the mean at time $k+1$ is:

$$\mathbb{E}\left(\overline{x}(k+1)|x(k)\right) = (1-r)\overline{x}(k)+\frac{r}{n},$$

which gives the first and second lines of (4.6).

Now, we can apply the same reasoning for the mean square value:

$$\begin{aligned}
n\overline{x^2}(k+1) &= \left((1-r)\left(1-\frac{1}{d_{out,i}(k)}\right)x_i(k)+\frac{r}{n}\right)^2+\left((1-r)\left(x_j(k)+\frac{1}{d_{out,i}(k)}x_i(k)\right)+\frac{r}{n}\right)^2\\
&+\sum_{h=1,h\neq i,j}^{n}\left((1-r)x_h(k)+\frac{r}{n}\right)^2\\
&= \left((1-r)^2\left(1-\frac{1}{d_{out,i}(k)}\right)^2x_i^2(k)+\frac{r^2}{n^2}+\frac{2r}{n}(1-r)\left(1-\frac{1}{d_{out,i}(k)}\right)x_i(k)\right)\\
&+\left((1-r)^2\left(x_j^2(k)+\frac{1}{d_{out,i}^2(k)}x_i^2(k)+\frac{2}{d_{out,i}(k)}x_j(k)x_i(k)\right)+\frac{r^2}{n^2}\right.\\
&\left.+\frac{2r}{n}(1-r)\left(x_j(k)+\frac{1}{d_{out,i}(k)}x_i(k)\right)\right)\\
&+\sum_{h=1,h\neq i,j}^{n}\left((1-r)^2x_h^2(k)+\frac{r^2}{n^2}+\frac{2r}{n}(1-r)x_h(k)\right).
\end{aligned}$$

Taking the expectation to the left and to the right of the equality gives:

$$\begin{aligned}
n\mathbb{E}\left(\overline{x^2}(k+1)|x(k)\right) &= \left((1-r)^2\left(1+\frac{1}{D}-\frac{2}{d}\right)\overline{x^2}(k)+\frac{r^2}{n^2}+\frac{2r}{n}(1-r)\left(1-\frac{1}{d}\right)\overline{x}(k)\right)\\
&+\left((1-r)^2\left(1+\frac{1}{D}\right)\overline{x^2}(k)+\frac{2}{d}\overline{x}^2(k)+\frac{r^2}{n^2}+\frac{2r}{n}(1-r)\left(1+\frac{1}{d}\right)\overline{x}(k)\right)\\
&+(n-2)\left[(1-r)^2\overline{x^2}(k)+\frac{r^2}{n^2}+\frac{2r}{n}(1-r)\overline{x}(k)\right]\\
&+\mathbb{E}\left(x_i(k)xj(k)-\overline{x}^2(k)\right).
\end{aligned}$$

For continuing this proof, we will use Hypothesis 4.2.1. We will speak about this simplification later.

$$n\mathbb{E}\left(\overline{x^2}(k+1)|x(k)\right) = (1-r)^2\left(n+\frac{2}{D}-\frac{2}{d}\right)\overline{x^2}(k)+\frac{2}{d}\overline{x}^2(k)+2r(1-r)\overline{x}(k)+n\frac{r^2}{n^2}$$

which gives the third line of (4.6) and ends the proof. □

Note also that, as we will see in the next section, pages are removed or added following always the same process, *i.e.*, the property explaining that an agent points towards another one stays the same over time. For this reason, d and D do not depend on k.

4.2.2 Replacement

The Randomized PageRank presents a decentralized method for computing the popularity of web pages. Moreover, it requires at each time small computations and does not need to know the entire structure of the considered network. However, this algorithm does not cope with modifications in the structure of the graph. As said in Subsection 4.1.2, the web contains a huge amount of pages. These pages can be modified so that it is very important to propose a method where the algorithm takes into account the modifications in the structure of the graph. In this subsection, we keep results of the Randomized PageRank algorithm for gossips, and we add a definition for replacements such that a web page can be exchanged with another one, *i.e.*, a page can be replaced.

In the previous chapters, a change in the value of a unique agent was synonym of replacement (Definition 1.0.2). From now on, replacing a web page will not only modify its popularity, but also change the structure of the network. In-links and out-links of the leaving web page will be deleted. Defining replacement is a task that can be achieved in different ways. Some methods will be explained in Appendix B. The best definition depends on what we are looking for: some methods seem more logical but do not guarantee stability for their steady state equation while others seem less efficient but converge toward a fixed point. By stability, we mean that the variance tends towards a fixed point. Thus, methods that do not ensure stability give a variance that tends towards infinity when we assume a number of time steps sufficiently large.

Before explaining how the popularity of an agent replacing the one that has left the system will be defined, it is important to precise how the structure of the graph will change. When an agent is replaced, all its adjacent links are removed. Then the agent that replaced the one that left the system is pointing towards each agent following a given property. On the other hand, each other agent is pointing towards this new node with this same property. For example, if we are working with random graph (Definition 3.3.2) following Erdos-Renyi model, the new agent will point towards the other with probability $P_c \in [0, 1]$ while the other agents will point towards this new node with also probability P_c. If we are working with random graph following configuration model, then our graph always follows the same degree sequence, and then we randomly select the agents that are connected to the new one such that the degree sequence holds.

Suppose that web page i leaves the system and that $\mathcal{V}$ and $\mathcal{E}$ are the set of web pages and directed edges respectively. We define the In-neighbours set of i $\mathcal{I}_i(k)$, *i.e.*, the set of all the agents that point towards i at time k and the Out-neighbours set of i $\mathcal{O}_i(k)$, *i.e.*, the set of all agents that are pointed by i at time k.

First of all, if agent e enters the graph at time k, then its popularity is equal to zero so that:

$$x_e(k) = 0.$$

Now, we have to decide how to update the other agents such that our solution guarantees stability and respects constraint provided in Equation (4.2). A first idea would say that each agent different from the leaving one l updates its popularity following:

$$x_j(k+1) = \frac{x_j(k)}{1 - x_l(k)} \quad \forall j \in \mathcal{V} \setminus l.$$

But this solution provides a square mean value and a variance that are not interesting. Indeed, they are not converging towards a fixed point, but are growing.

Thus, we opt for replacements where only agents pointing towards the leaving one receive a positive compensation, *i.e.*,

$$x_j(k+1) = x_j(k) + \frac{1}{d_{in,l}(k)} x_l(k) \quad j \in \mathcal{I}_l(k), \tag{4.7}$$

where $d_{in,l}(k)$ is the number of in-links for l at time k. For this reason, this replacement is named *Positive compensations with null input PageRank.*

Proposition 4.2.2. *The Positive compensations with null input PageRank replacement that is defined in Equation* (4.7) *knowing that* $\mathbb{E}\left(d_{in,i}(k)\right) = d$ *and* $\mathbb{E}(d^2_{in,i}(k)) = D \quad \forall i \in \mathcal{V}, k \geq 0$ *gives the following steady state equation:*

$$\mathbb{E}X(k+1) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{2}{n} & \frac{1}{n}\left(n-1+\frac{d}{D}\right) \end{pmatrix} \mathbb{E}X(k). \tag{4.8}$$

Note that d is also the expected value for the number of out-links of an agent that we defined in Proposition 4.2.1. Indeed, the number of out-links is equal to the number of in-links. Thus, their expected value is the same.

Proof. With this method, we obtain the following behaviour:

$$\begin{aligned} n\mathbb{E}\left(\overline{x}(k+1)|x(k)\right) &= (n-1-d)\overline{x}(k) + d\left(1+\frac{1}{d}\right)\overline{x}(k) \\ &= \overline{x}(k)(n-1-d+1+d) = n\overline{x}(k). \end{aligned}$$

Note that the expectation over $\overline{x}(k+1)$ is not necessary. Indeed, at each time step the mean of our system stays the same since on the one hand, we subtract the popularity of the leaving page. On the other hand, we add it to the pages that were pointing towards the leaving one. Now, looking at the mean square value and knowing that $\mathbb{E}(d^2_{in,i}(k)) = D \quad \forall i \in \mathcal{V}$ and $\forall k$:

$$\begin{aligned} n\mathbb{E}\left(\overline{x^2}(k+1)|x(k)\right) &= (n-1-d)\overline{x^2}(k) + d\left(\overline{x^2}(k) + \frac{2}{d}\overline{x}^2(k) + \frac{\overline{x^2}(k)}{D}\right) \\ &= \overline{x^2}(k)\left(n-1-d+d+\frac{d}{D}\right) + \overline{x}^2(k)d\frac{2}{d}. \end{aligned}$$

□

A last point that has to be clarified is: *How an agent knows that its neighbour has left the system?* To answer to this question, we can imagine that each agent is sending a request, *i.e.*, testing the proper functioning of a link after a small amount of time. If this period is sufficiently small, we can assume that it is equivalent to a situation where the leaving agent prevents its neighbours.

4.2.3 Discussion

Compared to the Classical PageRank, the Randomized PageRank with open systems offers two advantages. First of all, independently of the size of the network, the algorithm works and is decentralized. Obviously, the greater the graph is, the more we need time such that the popularity of each web page converges towards the value that the Classical PageRank would return. For instance, each time a gossip occurs in a complete graph, a web page has a probability $\frac{2}{|\mathcal{E}|}$ of being exchanging its value with another agent. Thus, assuming that the number of edges will growth with the number of agents: the larger n, the smaller the probability of being implied in the gossip is. This statement can be generalized for any graph. Secondly, Randomized PageRank with open systems also works with systems in which replacements occur. By definition of gossip and replacement, this method can cope with modifications in the structure of the graph. This point is very important since the content of a web page is something that evolves with time. Therefore, the links of a web page are susceptible to change. Note that in this document we assumed that the structure of a graph changes only when a replacement occurs. In other words, a gossip does not change the links of the network such that the gossip step presented in Equations (4.3), (4.4) and (4.5) converges towards the fixed point of the Equation (4.1). If the structure was always changing, then its fixed point would be always different. In this context, it is not possible to obtain popularities that are converging towards the fixed point of the Classical PageRank and thus, the scenario does not make sens.

4.2.4 Fixed point

Now that we have defined a replacement in the system, we can obtain the fixed point (Definition 1.1.1) for the variance thanks to Equations (4.6) and (4.8). In order to simplify our computation, remark that the mean $\overline{x}(k)$ is always equal to $\frac{1}{n}$ for all k by definition of Equations (4.6) and (4.8) and given our initial conditions. This allows us to solve a linear equation in function of $\overline{x^2}$ instead of a system of three equations with three unknowns.

Before going further, we will add a small hypothesis on the expected square value of the in-degree, *i.e.*, D of Proposition 4.2.2 which is described in Hypothesis 4.2.2.

Hypothesis 4.2.2. *The expected square value of the in-degree,* i.e., *D of Proposition 4.2.2 is equal to the square of the expected value of the out- and in-degree. It follows that:*

$$D = \mathbb{E}(d_{in,i}^2(k)) = \mathbb{E}(d_{in,i}(k))^2 = \mathbb{E}(d_{out,i}(k))^2 = d^2.$$

Note that Hypothesis 4.2.2 is equivalent to saying that the variance of the out- and in-degrees of the considered system is equal to zero.

We will in this section be focused on the fixed point a random graph where all the agents have the same out-degree and in-degree. A replacement occurs with probability p. Thus, we are in the context of Hypothesis 4.2.2, which is equivalent to the case of the configuration model where each entry of the degree sequence is the same. Obviously, the complete graph is a random graph using the configuration model where the degree sequences are made only of $n-1$ for each entry.

Proposition 4.2.3. *The fixed point (Definition 1.1.1) using Hypothesis 4.2.2 such that each entry of the degree sequences is equal to d and where a replacement occurs with probability p while a gossip occurs with probability $1-p$ is given by:*

$$\begin{cases} \mathbb{E}(\overline{x}^2)|_{eq} = \frac{1}{n}, \\ \mathbb{E}(\overline{x^2})|_{eq} = \frac{1}{n^2}, \\ \mathbb{E}(\mathrm{Var}(x))|_{eq} = \frac{(p-1)(\frac{-r^2+2r}{n^2}+\frac{2}{dn^3})+\frac{2p}{n^3}}{(p-1)(r-1)^2(1-\frac{2}{nd}+\frac{2}{nd^2})-\frac{p}{n}(1+\frac{1}{d(n-1)})(n-1)+1}. \end{cases} \tag{4.9}$$

The value of the fixed point for the mean square value is in the first instance not easy to interpret at all. To become more confident, we assume that $r = 0.3$, $p = 0.05$, $d = 20$ and $n = 21, \dots, 200$ which corresponds to reasonable scenarii. With these parameters we vary the value of n. In Figure 4.2 we can see that the fixed point of the square mean value quickly tends to 0 when n grows.

Now, if we look at the fixed point of the variance of the graph, we can observe in Figure 4.3 that it decreases quickly to zero. To obtain it, we keep the parameters chosen for the mean square value, and we just have to subtract $\frac{1}{n^2}$ to $\overline{x^2}$. These results fit with our expectations since we are using the configuration model where we impose an identical degree for each web page. Thus, the PageRank of each agent is always the same. Note also that, if $p = 0$, then the variance is always equal to zero. This result is logical since the time-average of the popularity of each agent when no replacement occurs tends towards the fixed-point of Equation (4.1).

4.2.5 Root mean square error (RMSE)

In this section we study the behaviour of the RMSE of the popularity of our web pages. The estimated vector is the one containing the popularity of our web pages obtained using the Randomized PageRank while the exact vector contains the popularity of our web pages using the Classical PageRank Algorithm. Obviously, if there is no replacement, we expect that the RMSE decreases until reaching 0 since we are in the case of the Randomized PageRank and the time-average of the value of an agent has to converge towards the fixed point of (4.1) [RFTI15]. On the other hand, after that a replacement occurred, we expect that the RMSE increases.

Definition 4.2.1. *The* ***Root Mean Square Error (RMSE)*** *is the standard deviation of an estimation compared to the exact value. In our case, the estimation will be the time-average of the popularity of our web pages using Randomized PageRank on open systems that we note $\hat{x}^{PR}$ while the exact value will be the popularity using the Classical PageRank, that we represent x^{PR}. It is defined as follows:*

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{x}_i^{PR} - x_i^{PR})^2}{n}}.$$

Since replacements occur with probability p, we have to define an efficient way such that the time-average (Definition 1.1.2) of the popularities of our web pages using Randomized PageRank with open systems do not take into account values that are too old, *i.e.*, values that exist before the replacement. Indeed, if the structure of the graph changes, then we do not want to keep values that are not matching with the new topology. A first idea would consist of defining a time-window (Definition 1.1.3) of length $\frac{1}{p}$ such that it follows in average the number of replacement. This method is very intuitive; however, it has the main disadvantage that if the number of time steps on which we consider the system is greater than $\frac{1}{p}$ ($tb >> \frac{1}{p}$) then our results do no longer fit with the replacement of the system since this method use the probability of a replacement for defining the length of the time-window. In practice, a replacement will not necessarily occur exactly at that moment. If we consider a huge number of time steps, then we can take into account values that do not fit at all with the current structure of the considered system.

For this reason, we opt for a time-window (Definition 1.1.2) whose size changes following the event that occurs. It evolves as follows:

$$\text{Size of time window} = \begin{cases} \text{Size of time window} + 1 & \text{if a gossip occurs} \\ 1 & \text{if a replacement occurs} \end{cases}.$$

This way has the main advantage that when a replacement occurs, it is very easy to visualize it in terms of $RMSE$. Moreover, the popularity that we assign to each web page is always related to the right graph structure.

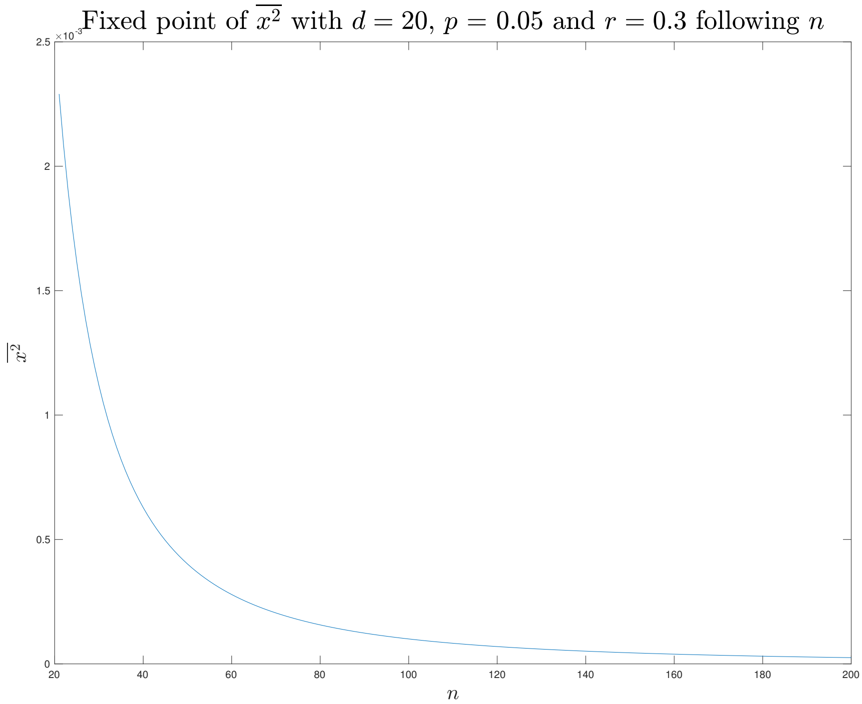


Figure 4.2: Fixed point for the mean square value following the number of agents which is between 21 and 200 using Equation (4.9).

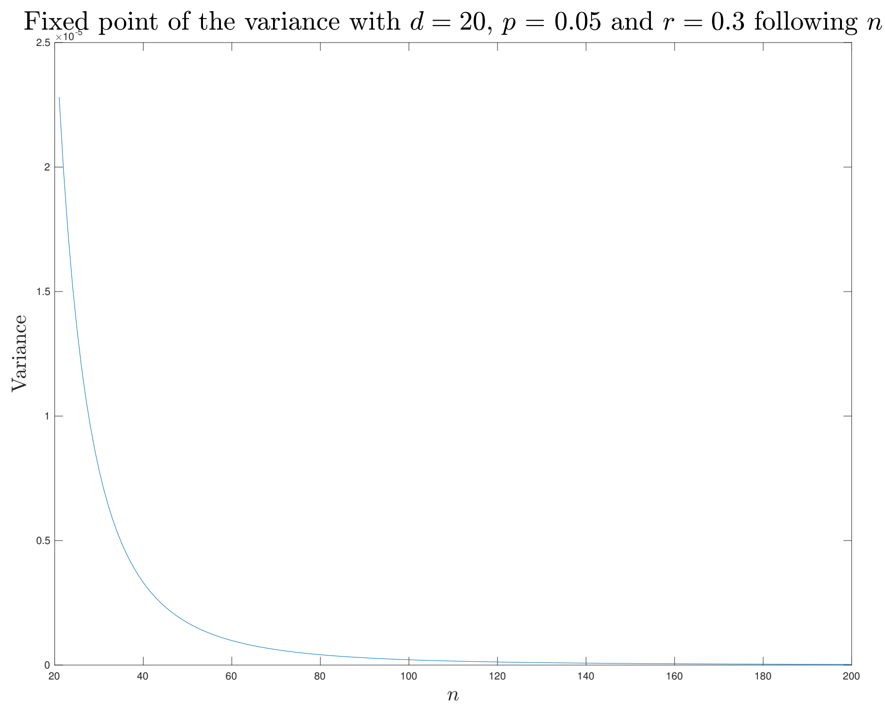


Figure 4.3: Fixed point for the variance following the number of agents which is between 21 and 200 using Equation (4.9).

RMSE on complete graphs

First of all, we will be working with complete graphs since they have the property that each web page has a PageRank equal to $\frac{1}{n}$. Indeed, since each web page is pointing towards all the others, all their contributions are the same. This structure has the advantage that we do not need to compute the popularity with the Classical PageRank. Thus, it is very easy to compute the RMSE.

In Figure 4.4, we can see the evolution of the root mean square error for a system made of 100 agents on 5 million time steps when no replacement occurs. Since we do not have to compute the popularity with the Classical PageRank, we are free to work with larger graphs. As we can see, the RMSE tends towards zero. It means that the popularity of all our web pages converges towards $\frac{1}{n}$ which is the expected PageRank for the complete graph. If we pay attention, we can see that for the first time steps of the graph, the RMSE starts from zero, then increases[1] and finally decreases. This phenomenon can be observed since we initialize the popularity of our web pages to $\frac{1}{n}$. Then the RMSE is initially equal to zero and increases with the first gossips. Note that the number of time steps is very large. This is for illustrating that the Randomized PageRank converges towards results of the classical one but slowly.

On the other hand, in Figure 4.5 is presented the RMSE for a system of 5 agents with probability of replacement equal to 0.0001. The number of agents is voluntarily small such that a replacement leads to a visible change in the RMSE of the system. We observe that the time window previously defined highlights pretty well the consequences of a replacement in the system. Indeed, we observe peaks for the RMSE just after a replacement occurred. After, we can see that the RMSE decreases but does not tend towards 0. The explanation for this phenomenon is that, as we can see in Figure 4.4, the RMSE decreases very slowly between two replacements. With a smaller probability of replacement p, the RMSE between two replacements should behave as it does in Figure 4.4 and tends towards 0. Note that, the larger p is, the larger the average value of the RMSE becomes since it does not have the time to converge towards zero.

Another way to be convinced about the fact that the RMSE grows with p consists in looking at the square of the RMSE. Since the square root is a monotonic function, considering the square of the RMSE does not modify the behaviour of the RMSE. Knowing that we are working with a complete graph and thus $x_i^{PR} = \frac{1}{n}$, we have:

$$\begin{aligned}
RMSE^2 &= \frac{\sum_{i=1}^{n} (\hat{x}_i^{PR} - x_i^{PR})^2}{n} \\
&= \frac{\sum_{i=1}^{n} \left(\left(\hat{x}_i^{PR}\right)^2 + \frac{1}{n^2} - \frac{2}{n}\hat{x}_i^{PR} \right)}{n} \\
&= \frac{n\overline{\left(\hat{x}_i^{PR}\right)^2} + \frac{1}{n} - 2\overline{\hat{x}_i^{PR}}}{n},
\end{aligned}$$

where in these equations, the same notation as in Definition 4.2.1 has been used. Now remember that whatever the event that occurs at a time step, the sum of all the popularity is always equal to 1. Thus, $\overline{\hat{x}_i^{PR}} = \frac{1}{n}$ and we have:

$$RMSE^2 = \overline{\left(\hat{x}_i^{PR}\right)^2} - \frac{1}{n^2}.$$

This equation is equal to 0 for the first time step, then it increases, and finally converges towards 0 if no replacement occurs. Thus, the square of the RMSE depends on the mean square value.

[1] during approximately the 2500 first time steps

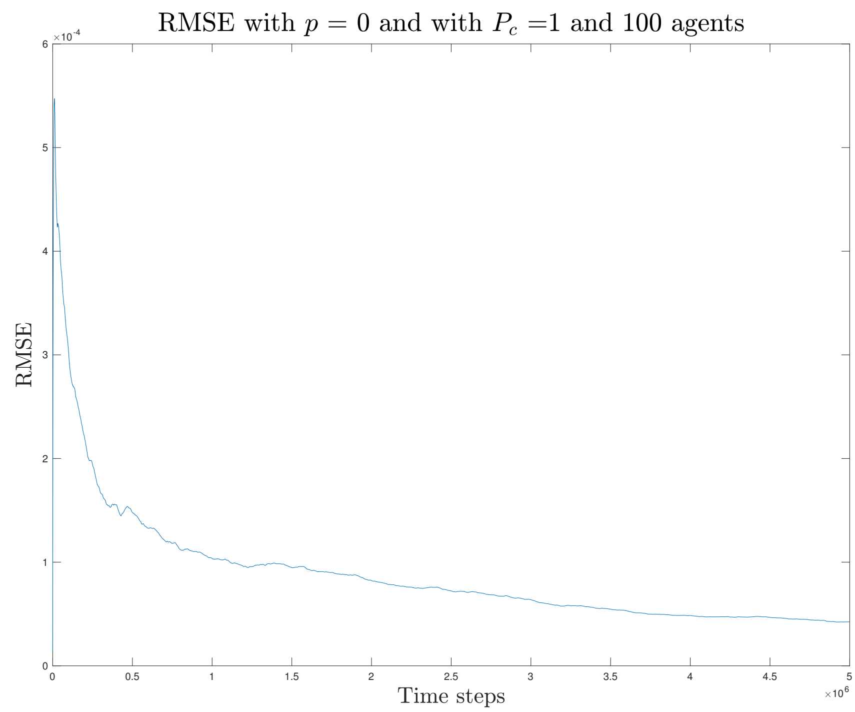


Figure 4.4: RMSE without replacement on 5 million time steps for a complete graph made of 100 agents.

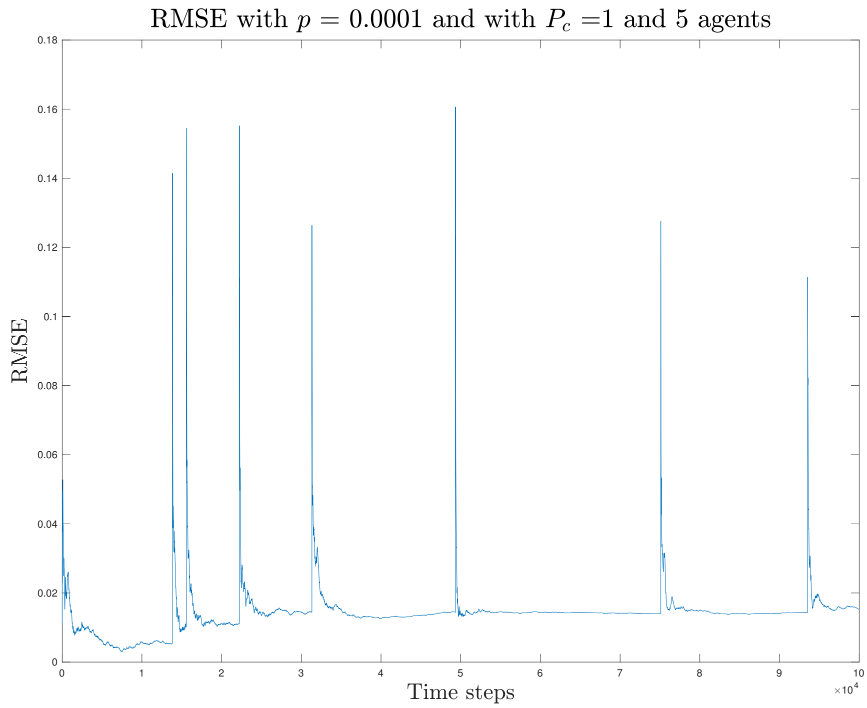


Figure 4.5: RMSE with a probability of replacement $p = 0.0001$ on 100000 time steps for a complete graph made of 100 agents.

Since we saw that the mean square value increases when p increases, we can assume that the statement above holds. In Figure 4.6, we can see the fixed point of the RMSE for $n = 100$ following the value of $p \in [0, 1]$. The growth of the fixed point of the RMSE highlights two things. First of all, the fact that the mean square value increases when p increases and secondly the fact that the RMSE is bigger when replacements occur more often.

About the convergence rate of the RMSE, we expect from [RFTI15] to have a convergence rate $\mathcal{O}(1/k)$. Obviously, the bigger the graph, the slower it converges. Indeed, if we consider only gossips, a web page will be implied in a gossip with a probability $\frac{2}{n}$ since we are in the case of a complete graph. In Figure 4.7, we can see the convergence rate of the RMSE in log-log scale. At the beginning the curve grows. This growth corresponds to the increasing values due to the initial condition we have chosen, *i.e.*, initializing all the popularities at $\frac{1}{n}$. Then it decreases following more or less a straight line. This behaviour fits with the expected convergence rate.

RMSE on random graphs

Now we extend results of the previous section for random graphs using Erdos-Renyi model. As we did previously, we introduce a variable P_c which is the probability that two web pages are connected. Since all the web pages have no longer the same popularity, we have to compute the "true" popularity using Classical PageRank each time a replacement occurs.

Another point which is important to remind is that, two web pages selected in a gossip are chosen as the two adjacent nodes of a randomly selected edge $e \in \mathcal{E}$. Thus the structure of our graph will influence the convergence rate of the RMSE towards 0. Agents that are less connected, *i.e.*, that have few In- and Out-links will be less gossiping than other web pages. For this reason, the RMSE will reach values that are bigger than the one obtained for complete graphs.

In Figure 4.8, we can see the evolution of the RMSE for a random graph of 10 agents with $P_c = 0.4$ when a replacement occurs with probability $p = 0.001$. The behaviour is the same as the one observed for complete graph. It is not surprising since the Randomized PageRank converges towards results of the Classical PageRank whatever the structure of the graph [RFTI15].

We can thus conclude that as we could expect, the RMSE decreases and tends towards zeros when gossips occur. When a replacement happens in the graph, the root mean square value increases sharply and the time-window (Definition 1.1.3) on which we are working is reset to one. Of course, the more often replacements occur, the more the fixed point of the RMSE is large.

4.2.6 Simulations

In this section we illustrate results obtained in the previous sections. We work on random graphs with the Erdos-Renyi model (Definition 3.3.2), defined by only one parameter: the probability P_c that agent i points to j, where i and j are randomly selected.

Gossip

First of all, we will look at the behaviour of our system when only gossips can occur, *i.e.*, when $p = 0$. We will consider a system made of $n = 50$ web pages with initial popularity equal to $\frac{1}{n} \quad \forall i = 1, \ldots, n$.

In Figures 4.9 and 4.10, we can see the variance and the square mean value of our network respectively. These results confirm the expected behaviour obtained in the previous sections despite some small numerical errors[2] in Figure 4.10.

Randomized PageRank with open systems also agrees with Conjecture 3.0.1 when we are working with different values of P_c. Indeed, as shown in Figures 4.11 and 4.12, the variance for

[2]probably due to `Matlab` and are about 10^{-16} large

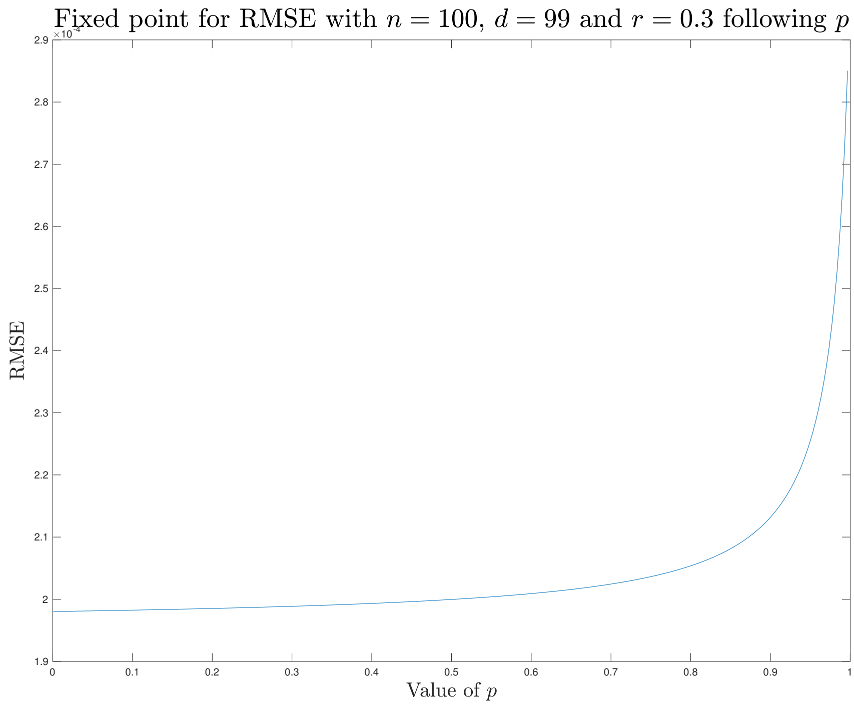


Figure 4.6: Fixed point of the RMSE following the probability of replacement p for a complete graph made of 100 agents.

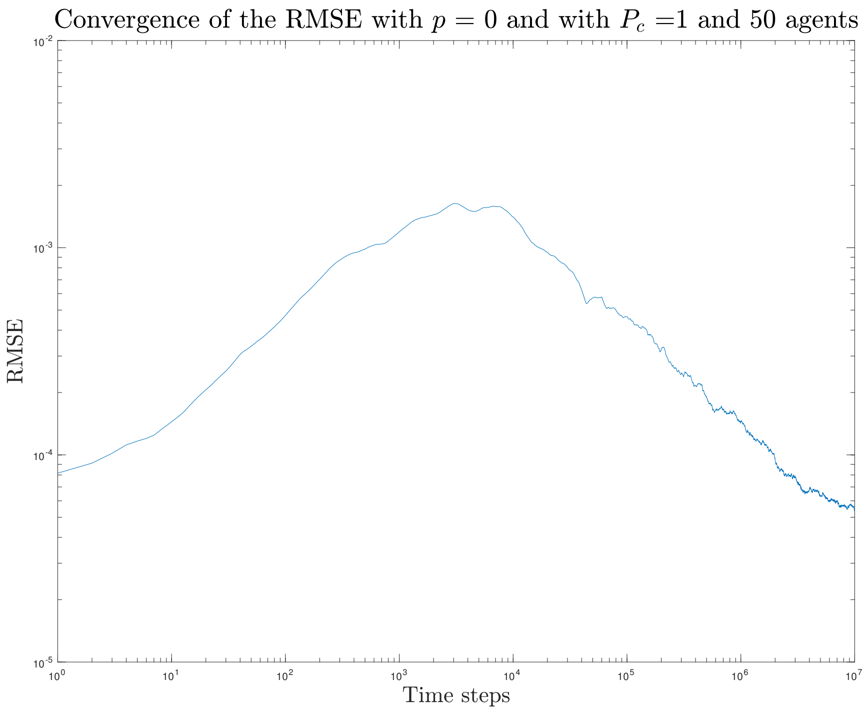


Figure 4.7: Convergence of the RMSE over 10^7 time steps for a complete graph made of 50 agents.

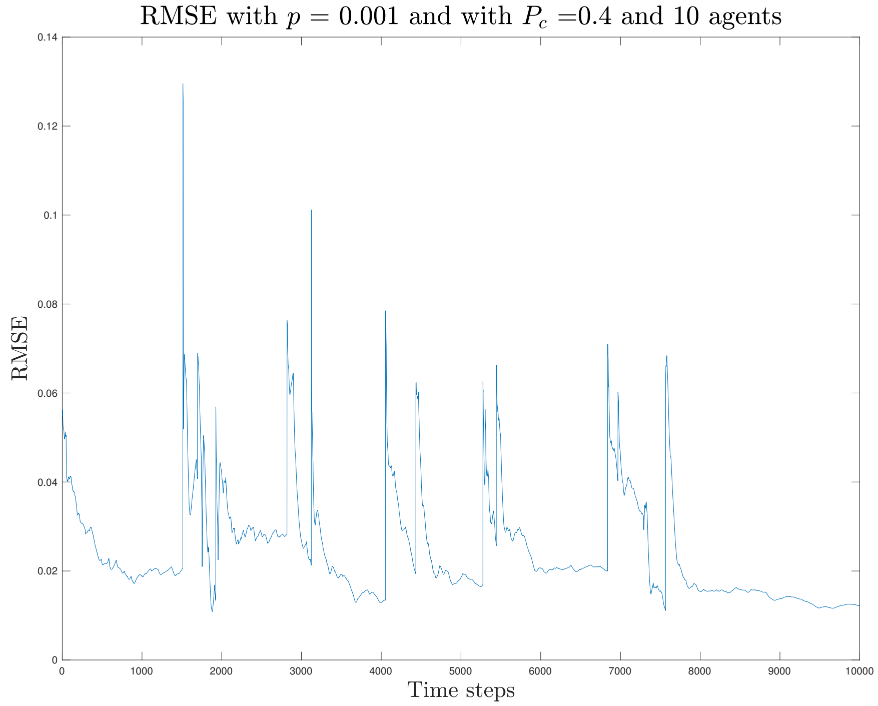


Figure 4.8: RMSE with a probability of replacement $p = 0.001$ on 10000 time steps and with a random graph using Erdos-Renyi model (Definition 3.3.2) with $P_c = 0.4$.

the complete graph, which is not very interesting for obtaining the popularity of web pages since the solution is obvious, is a lower bound for the variance of other graphs. The more P_c is small, the more the gap between the expected variance and the one obtained is large.

Under Hypothesis 4.2.2, we can see in Figure 4.13 the fixed point of the variance of this graph following the degree of our agents. This curve confirms our intuition: the larger d, the smaller the variance is.

Replacement

As in the previous part, let us consider a system made of 50 web pages where only replacements occur. Of course, for such a system we are not interested in the PageRank of our agents since they will never gossip, *i.e.*, they will never update their popularity in order to reach a value corresponding to the popularity they should have, given the structure of the graph. In our case, we are working with a complete graph such that $P_c = 1$. Note that, since only replacements happen, the structure of the graph has no influence on the results we have obtained.

In Figures 4.15 and 4.14, we can see the evolution of the variance and the square mean value of our system respectively. As expected with Equation (4.8), we observe that the square mean value stays constant. Like in the gossip part, the time-average (Definition 1.1.2) of the square mean value does not fit perfectly with our results. Again, we suppose that it is due to numerical errors since it is about 10^{-16}. On the other hand, the time-average of the variance fits pretty well with the expected value.

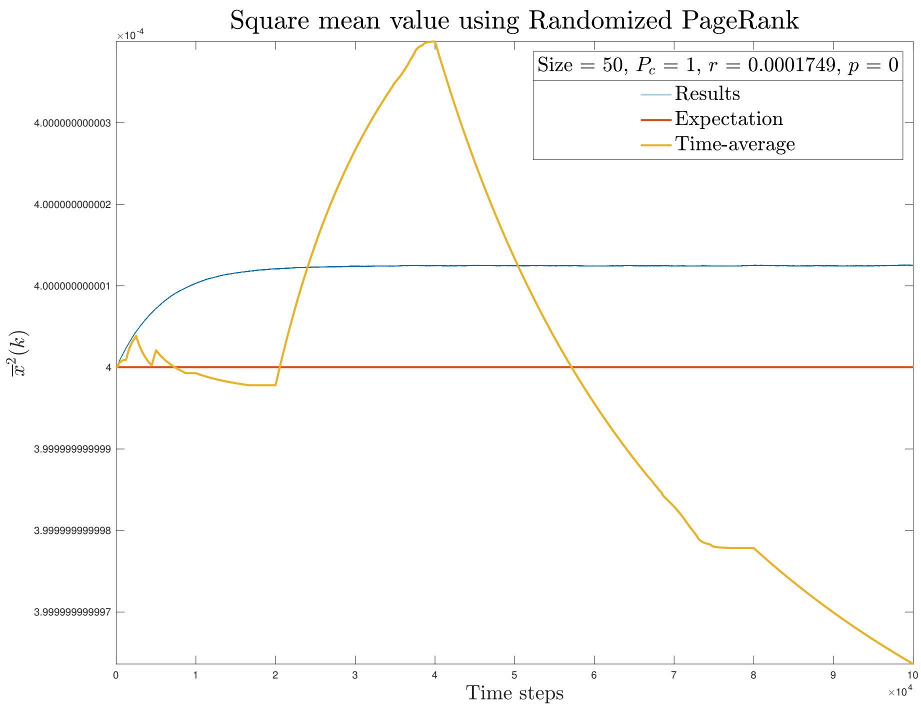


Figure 4.9: In red, the expected square mean value of the network using Randomized PageRank algorithm on 100000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 1$ using Equations (4.6) and (4.8) when no replacement occurs. In blue, the true square mean value (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

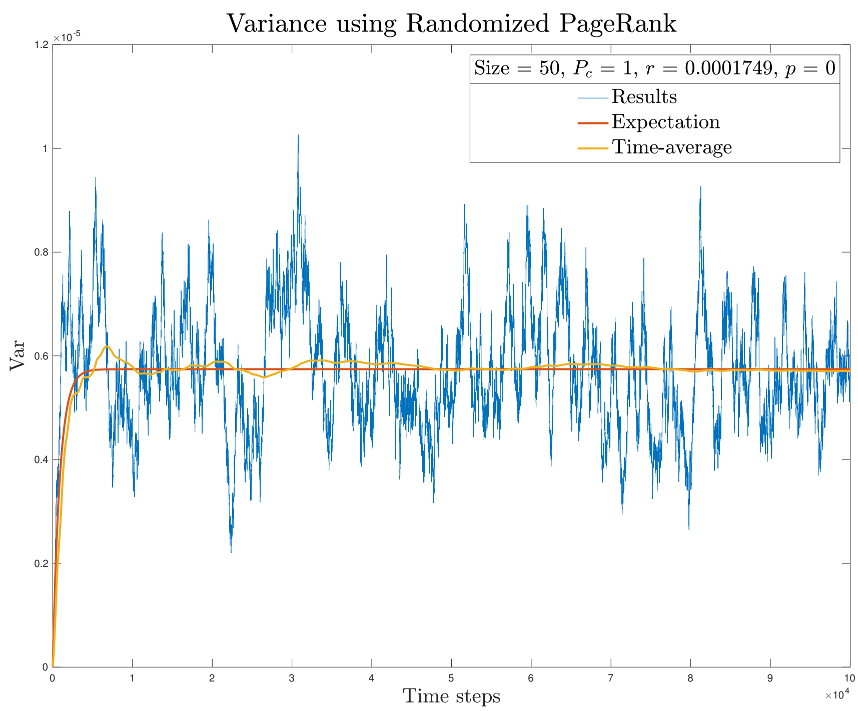


Figure 4.10: In red, the expected variance of the network using Randomized PageRank algorithm on 100000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 1$ using Equations (4.6) and (4.8) when no replacement occurs. In blue, the true variance (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

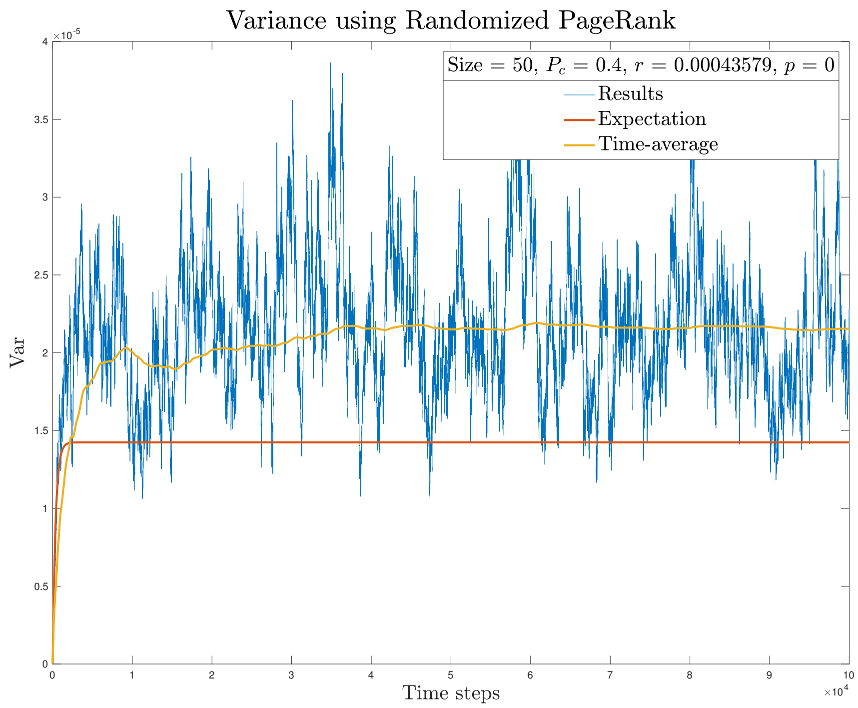


Figure 4.11: In red, the expected variance of the network using Randomized PageRank algorithm on 100000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 0.4$ using Equations (4.6) and (4.8) when no replacement occurs. In blue, the true variance (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

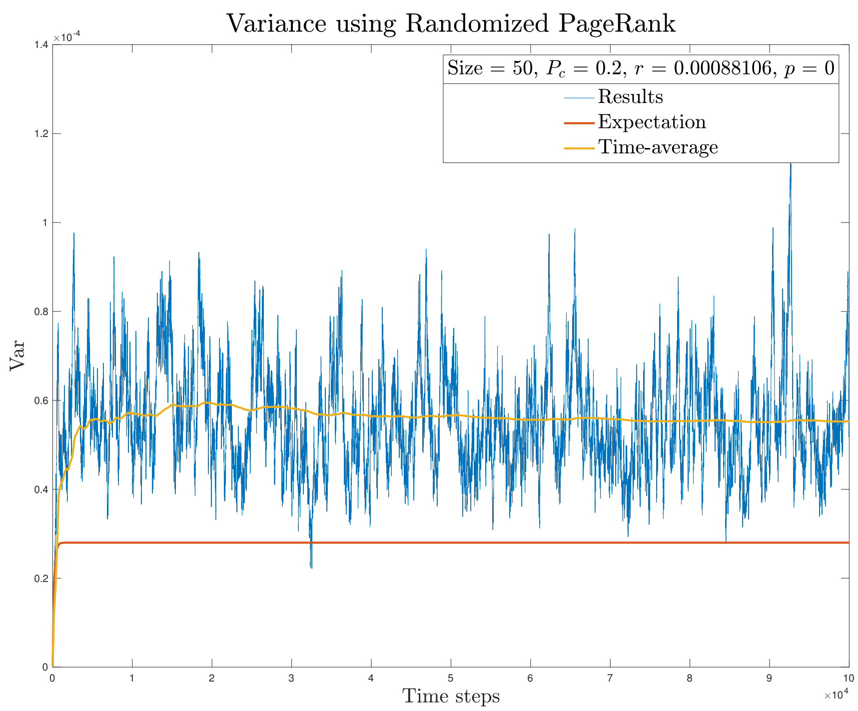


Figure 4.12: In red, the expected variance of the network using Randomized PageRank algorithm on 100000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 0.2$ using Equations (4.6) and (4.8) when no replacement occurs. In blue, the true variance (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

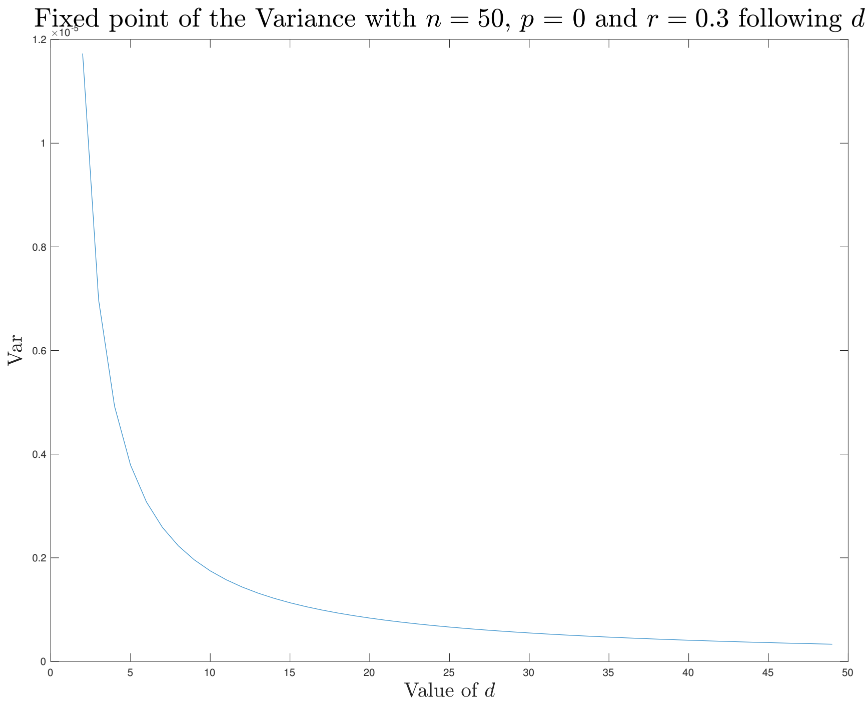


Figure 4.13: Fixed point of the variance following the value of d for a graph under Hypothesis 4.2.2 using Equations (4.6) and (4.8).

Gossip and replacement

Finally, let us study what happens when both gossips and replacements with a given probability are considered.

Conclusions obtained when only gossips or replacements occur still hold: the expectation of the complete graph will behave like a lower bound when P_c is smaller than 1. This statement supports Conjecture 3.0.1. On the other hand, when $P_c = 1$, our results tends towards the expectation we have combining Equations (4.6) and (4.8).

In Figures 4.16 and 4.17, we observe a gap when the number of connection between web pages is less frequent (in our case $P_c = 0.1$). This phenomenon is probably due to the heuristic we made for the computation of the steady state equation when a gossip occurs saying that $\mathbb{E}(x_j(k)x_i(k)) = \overline{x}^2(k)$, *i.e.*, Hypothesis 4.2.1.

4.2.7 Popularity distribution

Now that we have compared our results in terms of mean square value and variance, we can now wonder, how efficient is the Randomized PageRank algorithm in order to obtain the web pages popularity. First of all we will study the case where there is no replacement. Then we will extend our researches in order to support both gossips and replacements.

Popularity without replacement

In Figure 4.18, we can see the popularity of pages after 100000 time steps. Compared to the one returned with the Classical PageRank algorithm, values returned the Randomized PageRank respect in general the true ranking while smoothing a little bit extremely large and small values. However, we can conclude that the method is convincing since it works for large systems which

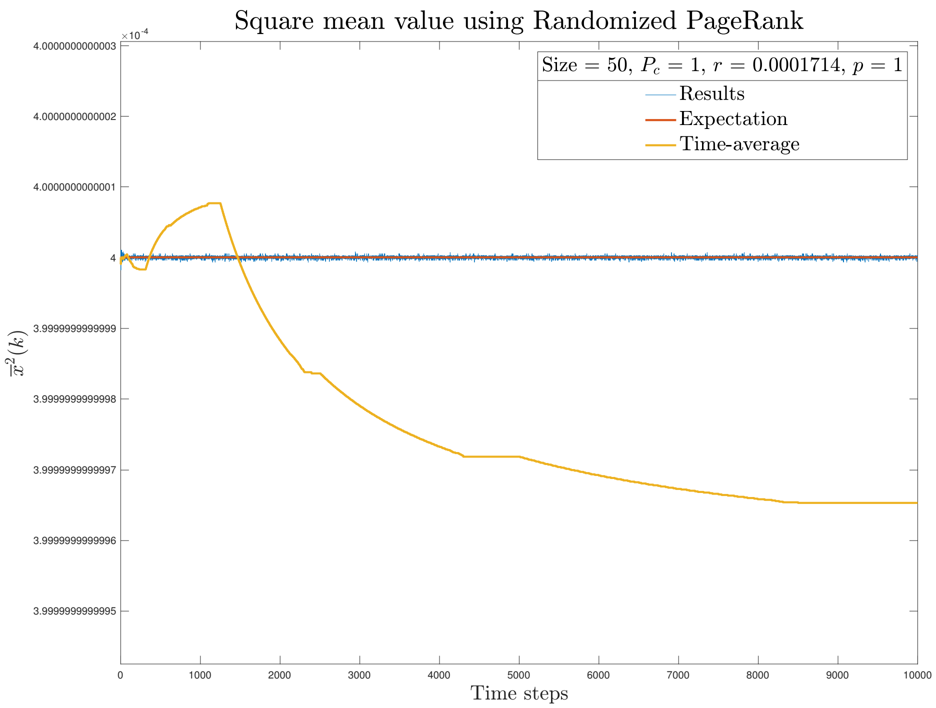


Figure 4.14: In red, the expected square mean value of the network using Randomized PageRank algorithm on 10000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 1$ using Equations (4.6) and (4.8) when only replacements occur. In blue, the true square mean value (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

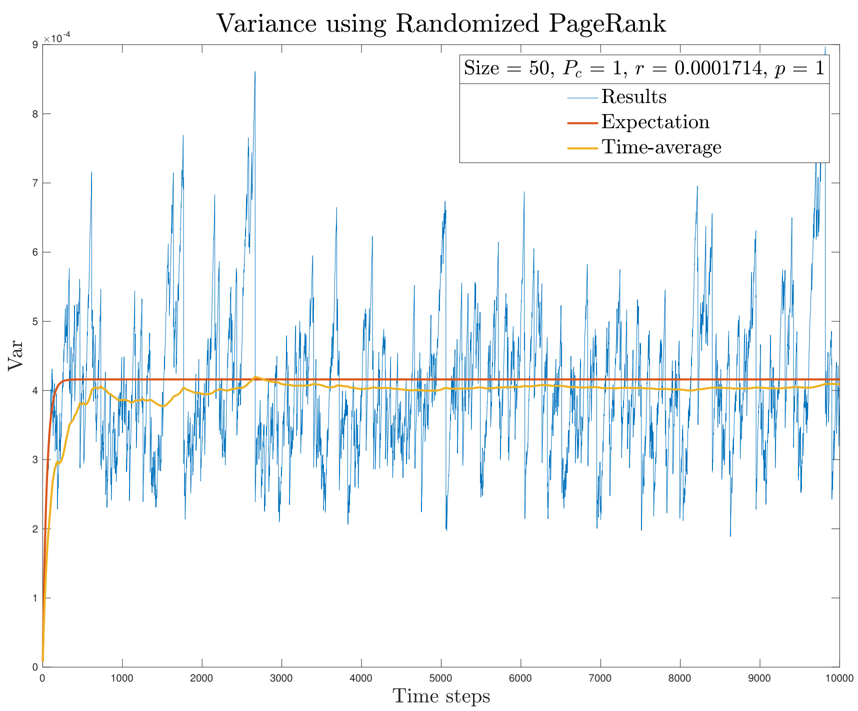


Figure 4.15: In red, the expected variance of the network using Randomized PageRank algorithm on 10000 time buckets with a random graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 1$ using Equations (4.6) and (4.8) when only replacements occur. In blue, the true variance (see Notation) of this system at each time step. In yellow, the time-average (Definition 1.1.2) of the blue curve.

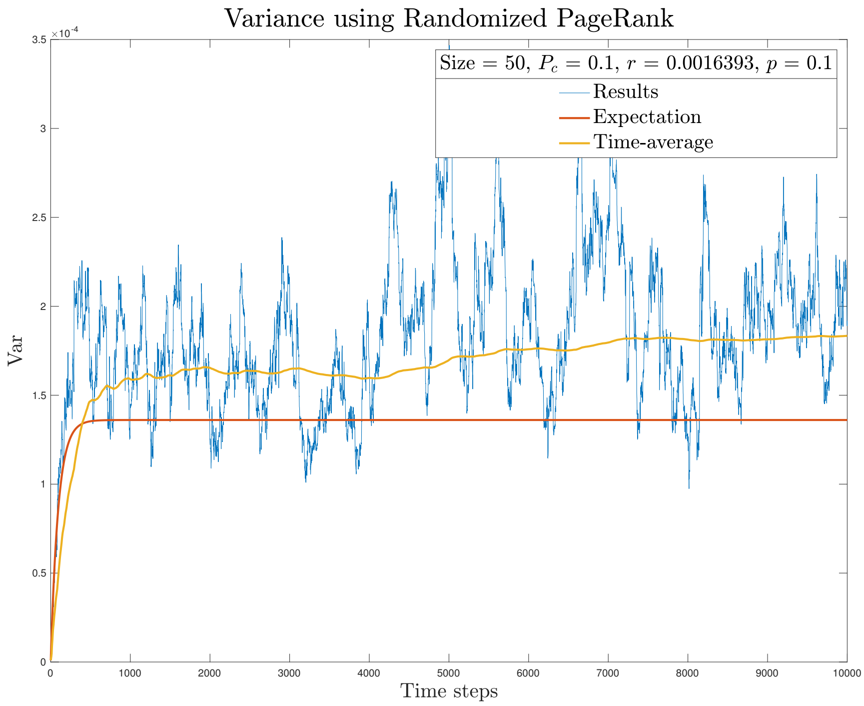


Figure 4.16: In red, the expected variance using Randomized PageRank algorithm on a graph following Erdos-Renyi model (Def. 3.3.2) with $P_c = 0.1$ using Equations (4.6) and (4.8) when the probability of a replacement is $p = 0.1$. In blue, the true variance (see Notation) of this system. In yellow, the time-average (Definition 1.1.2) of the blue curve.

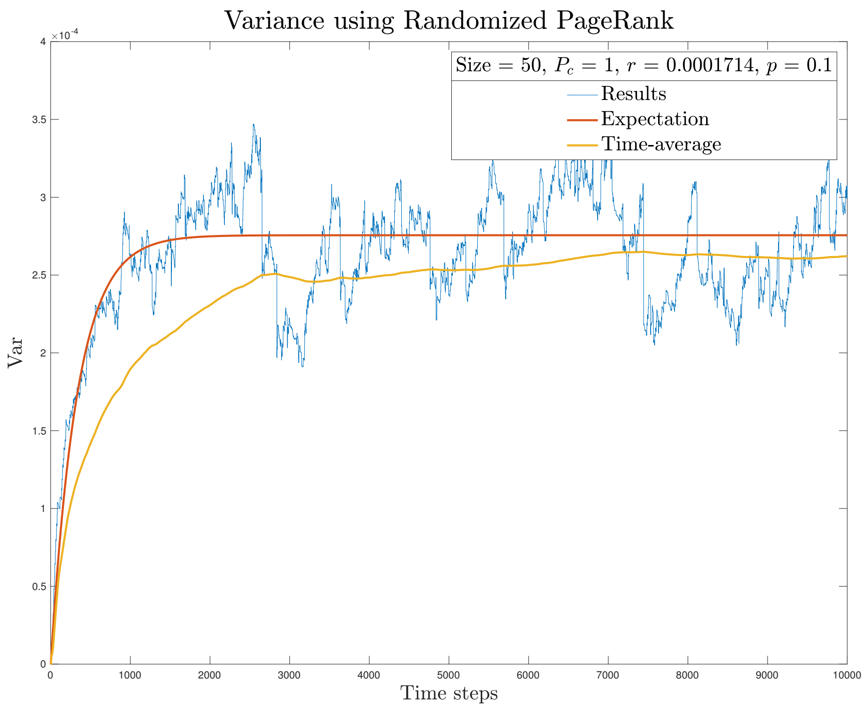


Figure 4.17: In red, the expected variance using Randomized PageRank algorithm on a graph following Erdos-Renyi model (Def. 3.3.2) with $P_c = 1$ using Equations (4.6) and (4.8) when the probability of a replacement is $p = 0.1$. In blue, the true variance (see Notation) of this system. In yellow, the time-average (Definition 1.1.2) of the blue curve.

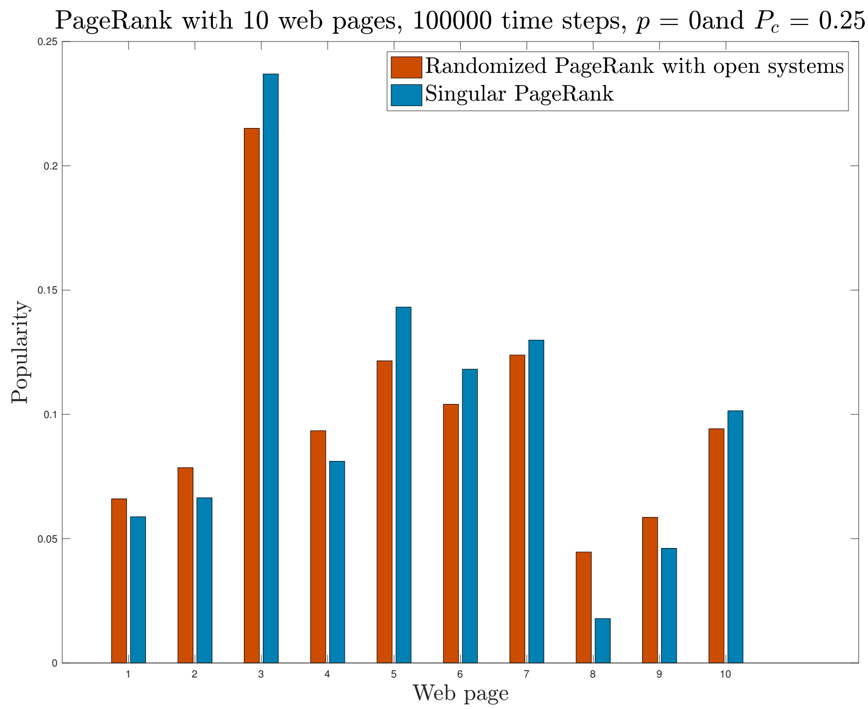


Figure 4.18: PageRank using the Classical algorithm vs. the Randomized one on a graph following Erdos-Renyi model (Definition 3.3.2) with $P_c = 0.25$ and with $n = 10$ where no replacement occurs.

is not the case for the Classical PageRank. Moreover, the important thing is the ranking of our pages not the value of their popularity.

Popularity with replacement

In Figure 4.18, we can see the popularity distribution for a random graph when there is no replacement. In this section, we will care about the PageRank of a complete graph where a replacement occurs with probability $p \in (0, 1]$. The idea is to keep the popularity vector just before a replacement occurs, and to compare this vector with the one we would obtain with the Classical PageRank method. We expect that both vectors behave more or less the same since between two replacements, only gossips can happen so that our system should behave like a system where no replacement occurs on a given time period. Obviously, the bigger p is, the less accurate the Randomized PageRank becomes.

In Figure 4.19, we have the RMSE of a system made of 10 agents following Erdos-Renyi model (Definition 3.3.2), where two web pages are connected with $P_c = 40\%$ of probability. A replacement occurs with probability $p = 10^{-7}$.

Now, in Figure 4.20, we can compare the popularity obtained for each web page using Classical PageRank and Randomized PageRank with open systems. These values for the popularity of our web pages have been selected just before the first replacement of the system occurs. As in Figure 4.18, Randomized PageRank smooths the values compared to Classical PageRank. This phenomenon can be explained since we initialize the popularity of each page at $\frac{1}{n}$ and the algorithm converges very slowly. Even if there is a difference between the two methods, results are pretty similar. This is because p is very small and then the network has in theory a lot of time steps before a replacement occurs. If we choose a bigger p, we expect to obtain more significant differences between the two algorithms.

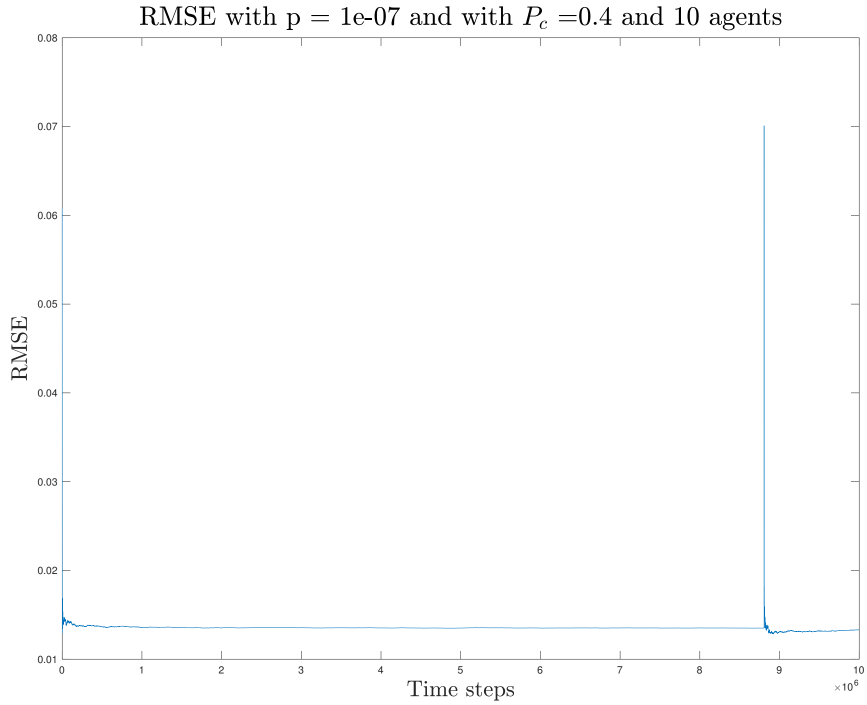


Figure 4.19: $RMSE$ with replacement for a random graph following Erdos-Renyi model (Definition 3.3.2) of 10 agents with $P_c = 0.4$ and with probability of replacement $p = 10^{-7}$.

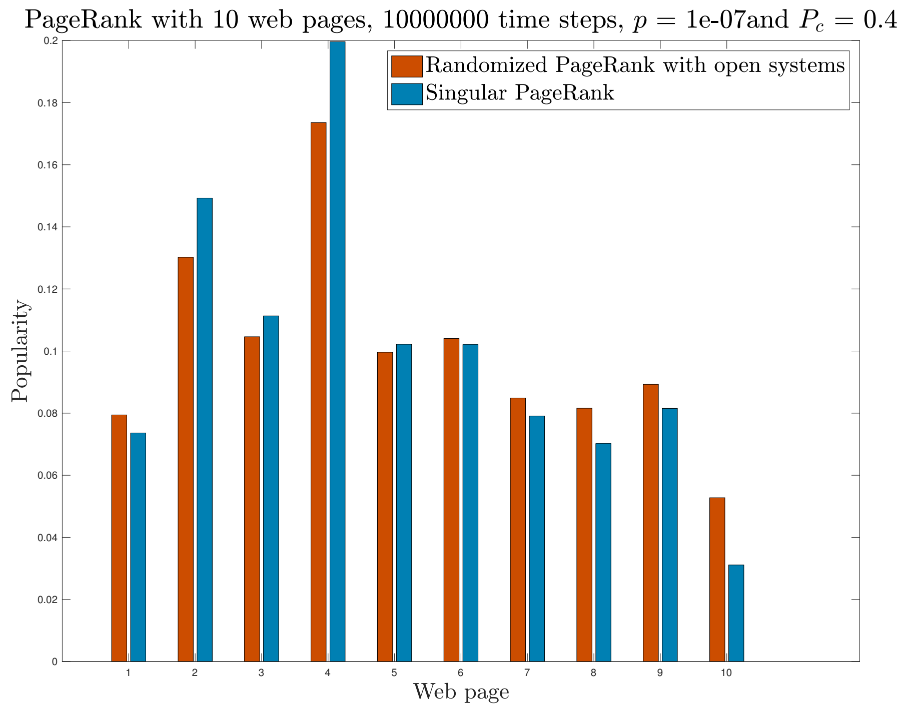


Figure 4.20: Comparison between Classical and Randomized PageRank for a random graph following Erdos-Renyi model (Definition 3.3.2) of 10 agents with $P_c = 0.4$ and with probability of replacement $p = 10^{-7}$.

4.2.8 Conclusion

Through this section, we presented a new way for computing the popularity of web pages in a directed graph. This algorithm has the advantages that it can adapt itself to modifications in the structure of the network. Moreover, it can also work with graphs that have a huge number of web pages.

Due to the use of Hypothesis 4.2.1 for the computation of the steady state equation when only gossips occur, we observe a gap between our results and the expected results for the variance of our system. Despite this phenomenon, we have decided to keep it as it is in order to have a steady state equation as readable as possible. We can also say that the Conjecture 3.0.1 seems to be still usable given the evolution of the fixed point following the expected out- and in-degree, d.

Concerning the convergence towards the values of the Classical PageRank, we can say that our algorithm works, although it smooths the popularity of web pages that have a very small or very large PageRank with the Classical algorithm. It can be explained by two things. First of all, because our algorithm converges very slowly. Secondly, by definition of our initial values, *i.e.*, the initialization of all the popularities to $\frac{1}{n}$.

Finally, we can affirm that the algorithm works pretty well despite a convergence rate that could be better. For this reason, researches about the way replacement are defined can still be done. It would be also interesting to try to define new update equations for gossips, for example by trying to imply more than two agents.

Conclusion

Open multi-agent systems and more generally distributed artificial intelligence are more and more required nowadays in computational problems. With the growing use of big data and large scale solution requirements, there is a good chance that researches about singular agent become rarer. On the other hand, distributed methods for problems solving turn out to be an interesting direction since they cope with two important points. First of all, they do not require a centralized data center and each agent is holding a part of the information contained in the system. Thus, systems we consider are able to react to modifications. Secondly, they work for very huge problems since they require, at each time and from each unit, a small amount of computation.

Through this master thesis, we have developed results for open multi-agent systems where a gossip or a replacement occurs at a time step with given probability. First of all, we continued the work conducted in [HM17] by testing, on systems with a complete graph structure, gossips between more than two agents with probability decreasing proportionally to the growth of the number of agents. We have seen that on the one hand, gossips between two agents are not those which converge the fastest. On the other hand, taking the number of agents implied in a gossip as the number of agents in the system is not either the best choice. For instance, in our simulations we observe that for a system of 50 agents, the variance of the values held by the agents converges very quickly when the number of agents in the gossip is equal to 10. We have also investigated the difference between the arithmetic mean and the alpha-mean (Definition 1.3.1) for gossips between two agents. As we could guess, the more the weight of each agent implied in the gossip is near one half, the faster the variance of our system will converge towards its fixed point.

Chapter 2 was also built on work in [HM17]. However, we have considered systems with a bipartite graph structure. First of all, we have computed the steady state equation of such graphs following their two partitions. If our results were fitting with the expected behaviour provided by the steady state equation, the main difference compared to the first chapter was the equation of the fixed point of the variance, of the square mean value and of the mean square value. On the one hand, numerical values agree with the expected behaviour of such graphs. On the other hand, symbolic values are not possible to derive on the contrary to Chapter 1. Indeed, using `Mathematica`, polynomials with pretty high degree were obtained. It was thus not possible to interpret them. To cope with this problem, we made a study of the behaviour of the fixed point of the bipartite graph following the respective sizes of the partitions and the probability that a replacement occurs. We finally ended this chapter with the following observation: *Whatever the size of the partitions and the probability that a replacement occurs are, the fixed point of the variance of the values held by the agents of a bipartite graph will always be greater than the one of a system with complete graph structure with the same number of agents and probability of replacement.* This statement has been the motivation for the third chapter.

In Chapter 3, we have seen that any graph always has a variance above the one of the complete graph for the same number of agents. We have explored different ways for explaining this acknowledgement. First of all, we verified that the steady state equation of the complete graph when a gossip is between two agents, from [HM17], was valid only for this graph and not for other structures such as vertex-transitive graphs. With that research, we discovered that the variance of the cycle graph was farther from the variance of the complete graph than the variance of the utility graph for the same number of agents. This result encouraged us to link the variance of a graph with its connectivity. Thus, we decided to quantify the variance of a system with a given number of agents following its diameter. For graphs with a small diameter, we obtained a significant growth which was quite interesting since it allowed our research to stop at this point. Unfortunately, when we work with graphs that have a big diameter, we observed that *it is no longer possible to state that the larger the diameter of a graph, the bigger the variance of the values of its agents is.* We therefore eliminated this track. The second idea that has been explored consisted in using the eigenvalues of the expected transition matrix of the graph. This research direction has already been investigated in [BGPS06]. Nevertheless, we had to adapt results from this article since the definition of gossip was not the same as the one used in this master thesis. We have finally proved that the upper bound for the variance of the complete graph was a lower bound on the upper bound of any other graph. Moreover, since a system behaves more like its upper bound than its lower bound, this result is very interesting and can be considered as a point providing good confidence in the statement saying that the variance of the complete graph is the smallest variance whatever the structure of the graph we consider.

Finally, Chapter 4 gives an example of the applications of open multi-agent systems on the PageRank algorithm. In [RFTI15], the authors propose a new definition for gossips such that the time-average of the value held by each agent converges towards the popularity that a classical PageRank algorithm would return. We generalized the research made through [RFTI15] by providing a definition for replacements. To achieve this, we have tried some different ways with the only constraint that the sum of all the popularities is equal to 1. First of all by giving to the new agent, *i.e.*, the one replacing the agent that left the system, a popularity depending on the connection it created. The value of other agents would be updated receiving a positive compensation if they were pointing towards the agent that has left the system and a negative compensation if they were pointed by the agent that has left the system. Secondly, by giving a popularity equal to 0 to the agent coming into the system and a positive or negative compensation for other agents following the same rule as in the first method. Unfortunately, both methods had the issue that the variance of the system was not converging towards a fixed point. We concluded from these two methods that the ones that seem intuitive work efficiently for a very small number of time steps. However, after more time steps, they lack of interest. The method we have selected is one where the new agent has a PageRank equal to 0 while other agents can only receive a positive compensation when they were pointing towards the agent that has left the system. If this solution seems less intuitive than the two previous ones, it gives interesting results for the variance of the system that converges towards a fixed point quickly. Moreover, we made some tests in order to ensure the accuracy of this method using the root mean square error between our definition of the PageRank using open multi-agent systems, the Randomized PageRank, and the classical PageRank algorithm. Results obtained are quite positive because the root mean square error follows a convergence rate equal to $\mathcal{O}(\frac{1}{k})$.

Ultimately, the world of distributed artificial intelligence is still full of the unknown. In the coming decades, we can hope to see an important growth of the utility of such methods. Not only in the scope of scientific activities but also, in the habits of everyday life. Recently, we have seen the arrival of autonomous vehicles such as the *Pop Up* project of Airbus, Italdesign and Audi or the *E-Volo* drone from Volocopter that are able to interact between themselves. All these discoveries are going to change the daily life. We can thus be pretty confident about the future and the applications of topics such as open multi-agent systems. This work is a modest introduction to this topic. Many other tracks remain to explore. For example and in the context of this master thesis, it would be very interesting to try new definitions for the replacement in Chapter 4. We can also investigate further about Conjecture 3.0.1. Finally, in the context of the three first chapters, it would be also interesting to study how a system behaves when replacements always modify its structure.

Appendix A

Evolution of the Variance Following n

In this section, we will quantify the evolution of the variance of the value held by agents in a graph. Obviously, if a partition is made of only one agent, its variance will be equal to 0. For a small number of agents, it will also be smaller than the variance associated to the distribution that the value hold by an agent follows.

Here, the value of each agent is a random variable following a uniform distribution on $[\frac{-1}{2}, \frac{1}{2}]$ so $\sigma^2 = \frac{1}{12}$.

Figure A.1 shows how this variance evolved with the number of agents in the considered graph. In order to obtain values that are sufficiently reliable, we have taken the average value on 100 000 different simulations. As we can see, below 25 agents, the value is significantly smaller than σ^2. To cope with this gap, we will keep in memory the value of the variance for each number of agents until 300. When the size of the graph is greater than this threshold, we will use σ^2 since it fits with the average value obtained.

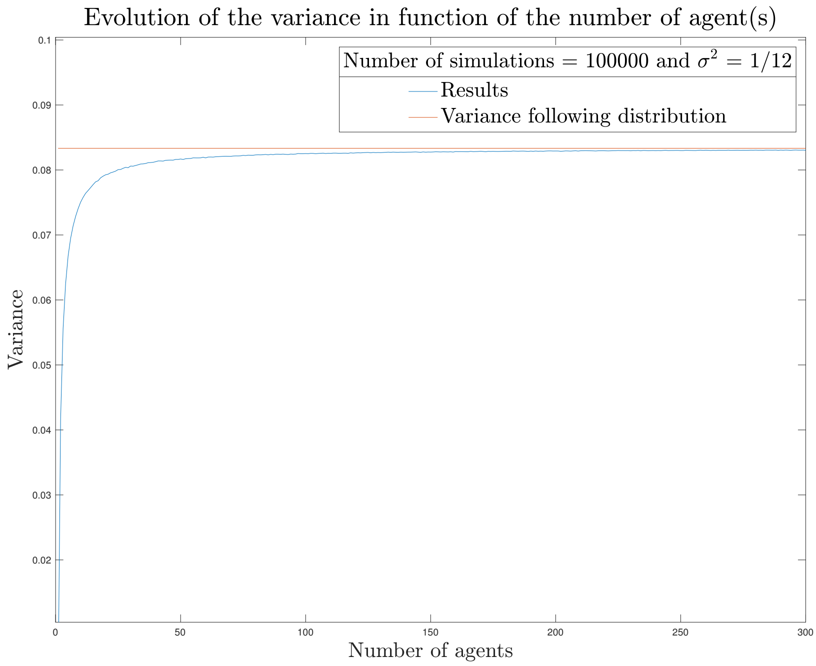


Figure A.1: Average variance σ^2 as a function of the number of agents for 100 000 simulations

Appendix B

Replacement for Randomized PageRank with Open Systems

There exist many possibilities for defining a replacement in an open system. The aim of this section is to present two of them that are different from the *Positive compensations with null input PageRank* replacement and to explain why they are not interesting in our case.

B.1 Positive/negative compensations with positive input PageRank

If agent l leaves at time k, we will quantify this departure by the following updates:

$$x_j(k+1) = \begin{cases} \left(1 + \frac{1}{d_{out,j}(k)}\right) x_j(k) & j \in \mathcal{I}_l(k) \\ x_j(k) - \frac{1}{d_{out,l}(k)} x_l(k) & j \in \mathcal{O}_l(k) \\ \left(1 + \frac{1}{d_{out,j}(k)}\right) x_j(k) - \frac{1}{d_{out,l}(k)} x_l(k) & j \in \mathcal{I}_l(k) \cap \mathcal{O}_l(k) \end{cases} \tag{B.1}$$

The first case of (B.1), *i.e.*, when $j \in \mathcal{I}_l(k)$ means that when a page is deleted or more generally removed from the system, the popularity that the pages pointing towards the removed one were giving is no more provided and thus, those pages will keep it for their own. The second case of (B.1) means that the pages that were pointed by the removed page do not benefit of this reference anymore. The third case is the intersection of the two first ones.

Suppose now that web page e is simultaneously added into the system. The first step consists in generating in-links randomly from some of the remaining agents and out-links from k towards the other web pages. In this context, we will assume that we are under Hypothesis 4.2.2 when adding e. The second step consists in assigning a popularity to e. To achieve this, several possibilities exists. We could first say that popularity of e is equal to the average of the popularity of agents pointing toward it. But this solution does not take into account the number of out-links associated to each agent having an in-link toward e. We could also decide to fix the popularity of e at one over the number of web pages of the system. This value corresponds to the initial value when using PageRank. A third possibility consists in defining the popularity of e at time $(k+1)^-$, *i.e.*, time just after the web page leaves the system, but just before e enters it, as follows:

$$x_e(k+1)^- = \sum_{j \in \mathcal{I}_e(k+1)^-} \frac{x_j(k+1)^-}{d_{out,j}(k+1)^-}$$

On the one hand, this alternative respects the PageRank policy of the in-links web pages. Moreover, it offers more degree of freedom than the second possibility where the popularity of the new agent is always the same.

On the other hand, using such a popularity for e will modify the sum of the PageRank of all the web pages, and it will no more be equal to 1, as we will see below. Suppose a replacement occurs in k, web page l leaves the system whilst e enters it and for any page i, its PageRank at time k is equal to $x_i(k)$:

$$\begin{aligned}
n\overline{x}(k+1) = & \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} x_i(k) \\
& + \sum_{\substack{i \in \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} \left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) + \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \in \mathcal{O}_l(k)}} \left(x_i(k) - \frac{1}{d_{out,l}(k)} x_l(k)\right) \\
& + \sum_{i \in \mathcal{I}_l(k) \cap \mathcal{O}_l(k)} \left(\left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) - \frac{1}{d_{out,l}(k)} x_l(k)\right) + \left(\sum_{i \in \mathcal{I}_e(k+1)^-} \frac{x_i(k+1)^-}{d_{out,i}(k+1)^-}\right) \\
& = \sum_{i \neq l} x_i(k) + \sum_{i \in \mathcal{I}_l(k)} \frac{x_i(k)}{d_{out,i}(k)} - \sum_{i \in \mathcal{O}_l(k)} \frac{x_l(k)}{d_{out,l}(k)} + \sum_{i \in \mathcal{I}_e(k+1)^-} \frac{x_i(k+1)^-}{d_{out,i}(k+1)^-}
\end{aligned}$$

Where $i \neq l$.
Taking the expectation of both sides of the equality gives:

$$\begin{aligned}
n\mathbb{E}\left(\overline{x}(k+1) | x(k)\right) &= (n-1)\overline{x}(k) + d\frac{\overline{x}(k)}{d} - d\frac{\overline{x}(k)}{d} \\
&+ d\frac{\overline{x}(k)}{d} \\
&= n\overline{x}(k)
\end{aligned}$$

Note that the reasoning above holds since

$$\begin{aligned}
\mathbb{E}\left(x_i(k+1)^- | x(k)\right) = & \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} x_i(k) \\
& + \sum_{i \in \mathcal{I}_l(k)} \left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) + \sum_{i \in \mathcal{O}_l(k)} \left(x_i(k) - \frac{1}{d_{out,l}(k)} x_l(k)\right) \quad i \neq l
\end{aligned}$$

Of course, if we repeat this process over a huge number of replacements, the sum of the popularity over the network can become different from 1. To cope with this modification, two possibilities exist. The first one would ignore this change, *i.e.*, relax constraint (4.2) while the second one would normalize the popularity of our web pages in order to keep (4.2) true. We will explain later which solution is the best one.

Now we can work on the evolution of the mean square value. We know that:

$$
\begin{aligned}
n\overline{x^2}(k+1) = & \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} x_i^2(k) \\
& + \sum_{\substack{i \in \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} \left(1 + \frac{1}{d_{out,i}(k)}\right)^2 x_i^2(k) + \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \in \mathcal{O}_l(k)}} \left(x_i(k) - \frac{1}{d_{out,l}(k)} x_l(k)\right)^2 \\
& + \sum_{i \in \mathcal{I}_l(k) \cap \mathcal{O}_l(k)} \left(\left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) - \frac{1}{d_{out,l}(k)} x_l(k)\right)^2 \\
& + \left(\sum_{i \in \mathcal{I}_e(k+1)^-} \frac{x_i(k+1)^-}{d_{out,i}(k+1)^-}\right)^2 \\
= & \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} x_i^2(k) + \sum_{\substack{i \in \mathcal{I}_l(k) \\ i \notin \mathcal{O}_l(k)}} \left(1 + \frac{1}{d_{out,i}^2(k)} + \frac{2}{d_{out,i}(k)}\right) x_i^2(k) \\
& + \sum_{\substack{i \notin \mathcal{I}_l(k) \\ i \in \mathcal{O}_l(k)}} \left(x_i^2(k) + \frac{1}{d_{out,l}^2(k)} x_l^2(k) - \frac{2}{d_{out,l}(k)} x_i(k) x_l(k)\right) \\
& + \sum_{i \in \mathcal{I}_l(k) \cap \mathcal{O}_l(k)} \left(\left(1 + \frac{1}{d_{out,i}^2(k)} + \frac{2}{d_{out,i}(k)}\right) x_i^2(k) + \frac{1}{d_{out,l}^2(k)} x_l^2(k)\right. \\
& \left. - \frac{2}{d_{out,l}(k)} \left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) x_l(k)\right) \\
& + \left(\sum_{i \in \mathcal{I}_e(k+1)^-} \frac{x_i(k+1)^-}{d_{out,i}(k+1)^-}\right)^2
\end{aligned}
$$

Where $i \neq l$.
Before taking the expectation of both sides of the equality, we have to think about the size of $\mathcal{I}$ and $\mathcal{O}$. As a reminder, we are under Hypothesis 4.2.2.

With this information, we can compute the expectation of the mean square value:

$$
\begin{aligned}
n\mathbb{E}\left(\overline{x^2}(k+1)|x(k)\right) = & (n - 1 - 2d + \frac{d^2}{n-1})\overline{x^2}(k) + (d - \frac{d^2}{n-1}) \left(1 + \frac{1}{d^2} + \frac{2}{d}\right) \overline{x^2}(k) \\
& + (d - \frac{d^2}{n-1}) \left(\overline{x^2}(k) + \frac{1}{d^2}\overline{x^2}(k) - \frac{2}{d}\overline{x}^2\right) \\
& + \frac{d^2}{n-1} \left(\left(1 + \frac{1}{d^2} + \frac{2}{d}\right) \overline{x^2}(k) + \frac{1}{d^2}\overline{x^2}(k) - \frac{2}{d}\left(1 + \frac{1}{d}\right)\overline{x}^2\right) \\
& + \mathbb{E}\left(\sum_{i \in \mathcal{I}_e(k+1)^-} \frac{x_i(k+1)^-}{d_{out,i}(k+1)^-}\right)^2
\end{aligned}
$$

With this expectation, we find the main issue of assigning such a PageRank to new page e. Indeed, we have no idea about how the square of the popularity of e is expected to be.

B.2 Positive/negative compensations with null input PageRank

In this alternative, equation (B.1) still holds but now the PageRank of the agent entering the system will be defined by the following:

$$x_e(k+1)^- = 0 \tag{B.2}$$

This solution is suitable since it will simplify our computation for the mean square value.

Proposition B.2.1. *With this initial value for the entering web page, the steady state equation is:*

$$\mathbb{E}X(k+1) = \frac{1}{n}\begin{pmatrix} n-1 & 0 & 0 \\ 0 & \frac{(n-1)^2}{n} & 0 \\ 0 & -2\left(1+\frac{1}{n-1}\right) & n-1+\frac{d}{n-1}+2 \end{pmatrix}\mathbb{E}X(k). \tag{B.3}$$

Proof. Let us compute how the mean will evolve after such a replacement occurs:

$$\begin{aligned}
n\overline{x}(k+1) = & \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} x_i(k) \\
& + \sum_{\substack{i \in \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} \left(1+\frac{1}{d_{out,i}(k)}\right) x_i(k) + \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \in \mathcal{O}_e(k)}} \left(x_i(k) - \frac{1}{d_{out,e}(k)} x_e(k)\right) \\
& + \sum_{i\in\mathcal{I}_e(k)\cap\mathcal{O}_e(k)} \left(\left(1+\frac{1}{d_{out,i}(k)}\right) x_i(k) - \frac{1}{d_{out,e}(k)} x_e(k)\right) \\
& = (n-1)\overline{x}(k) + \sum_{i\in\mathcal{I}_e(k)} \frac{x_i(k)}{d_{out,i}(k)} - \sum_{i\in\mathcal{O}_e(k)} \frac{x_e(k)}{d_{out,e}(k)}
\end{aligned}$$

Where $i \neq e$.
Taking the expectation of both sides of the equality gives:

$$\begin{aligned}
n\mathbb{E}\left(\overline{x}(k+1)\right) &= (n-1)\overline{x}(k) + d\frac{\overline{x}(k)}{d} - d\frac{\overline{x}(k)}{d} \\
&= (n-1)\overline{x}(k)
\end{aligned}$$

We can observe that using (B.2) the average mean will decrease between two time steps where a replacement occurs. Obviously, if we remove the PageRank of one web page without giving a positive compensation to the others, we will obtain a decreasing mean with time.
Now we can proceed as we did with the previous definition of x_e in order to obtain the mean square value of our system:

$$
\begin{aligned}
n \cdot \overline{x^2}(k+1) = & \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} x_i^2(k) \\
& + \sum_{\substack{i \in \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} \left(1 + \frac{1}{d_{out,i}(k)}\right)^2 x_i^2(k) + \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \in \mathcal{O}_e(k)}} \left(x_i(k) - \frac{1}{d_{out,e}(k)} x_e(k)\right)^2 \\
& + \sum_{i \in \mathcal{I}_e(k) \cap \mathcal{O}_e(k)} \left(\left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) - \frac{1}{d_{out,e}(k)} x_e(k)\right)^2 \\
= & \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} x_i^2(k) + \sum_{\substack{i \in \mathcal{I}_e(k) \\ i \notin \mathcal{O}_e(k)}} \left(1 + \frac{1}{d_{out,i}^2(k)} + \frac{2}{d_{out,i}(k)}\right) x_i^2(k) \\
& + \sum_{\substack{i \notin \mathcal{I}_e(k) \\ i \in \mathcal{O}_e(k)}} \left(x_i^2(k) + \frac{1}{d_{out,e}^2(k)} x_e^2(k) - \frac{2}{d_{out,e}(k)} x_i(k) x_e(k)\right) \\
& + \sum_{i \in \mathcal{I}_e(k) \cap \mathcal{O}_e(k)} \left(\left(1 + \frac{1}{d_{out,i}^2(k)} + \frac{2}{d_{out,i}(k)}\right) x_i^2(k) + \frac{1}{d_{out,e}^2(k)} x_e^2(k) \right. \\
& \left. - \frac{2}{d_{out,e}(k)} \left(1 + \frac{1}{d_{out,i}(k)}\right) x_i(k) x_e(k)\right)
\end{aligned}
$$

Where $i \neq l$.
Now, taking the expectation gives:

$$
\begin{aligned}
n \cdot \mathbb{E}\left(\overline{x^2}(k+1) | x(k)\right) = & (n - 1 - 2d + \frac{d^2}{n-1}) \overline{x^2}(k) + (d - \frac{d^2}{n-1}) \left(1 + \frac{1}{d^2} + \frac{2}{d}\right) \overline{x^2}(k) \\
& + (d - \frac{d^2}{n-1}) \left(\overline{x^2}(k) + \frac{1}{d^2} \overline{x^2}(k) - \frac{2}{d} \overline{x}^2\right) \\
& + \frac{d^2}{n-1} \left(\left(1 + \frac{1}{d^2} + \frac{2}{d}\right) \overline{x^2}(k) + \frac{1}{d^2} \overline{x^2}(k) - \frac{2}{d}\left(1 + \frac{1}{d}\right) \overline{x}^2\right) \\
= & \overline{x^2}(k) \left[n - 1 + \frac{d}{n-1} + 2\right] - 2\overline{x}^2(k) \left[1 + \frac{1}{n-1}\right]
\end{aligned}
$$

□

As we can see in (B.3), the main issue is that the mean and the square mean value will decrease after each replacement while the mean square value will always increase. This growth will not provide a stability for our system. For this reason, we will no more use (B.1).

Appendix C

Barbell Graph Behaviour Following n_2

This chapter aims to show two things. First of all, the gap between the square mean value of a barbell graph and the square mean value of a complete graph. Secondly, the gap between the variance of a barbell graph and the variance of a complete graph with the same number of agents. In Figures 3.4 and 3.5 that we presented in chapter 3, we show how those gaps behaves. In this section, we want to confirm conclusion about the behaviour that we draw.

Figures C.1 and C.3 show how $\overline{x}^2$ evolves when $n_2 = 50$ and $n_2 = 100$ while Figures C.2 and C.4 show how the variance changes when $n_2 = 100$ and $n_2 = 50$.

From these Figures, we conclude that the gap for the square mean value and the variance will behave as it does when $n_2 = 200$.

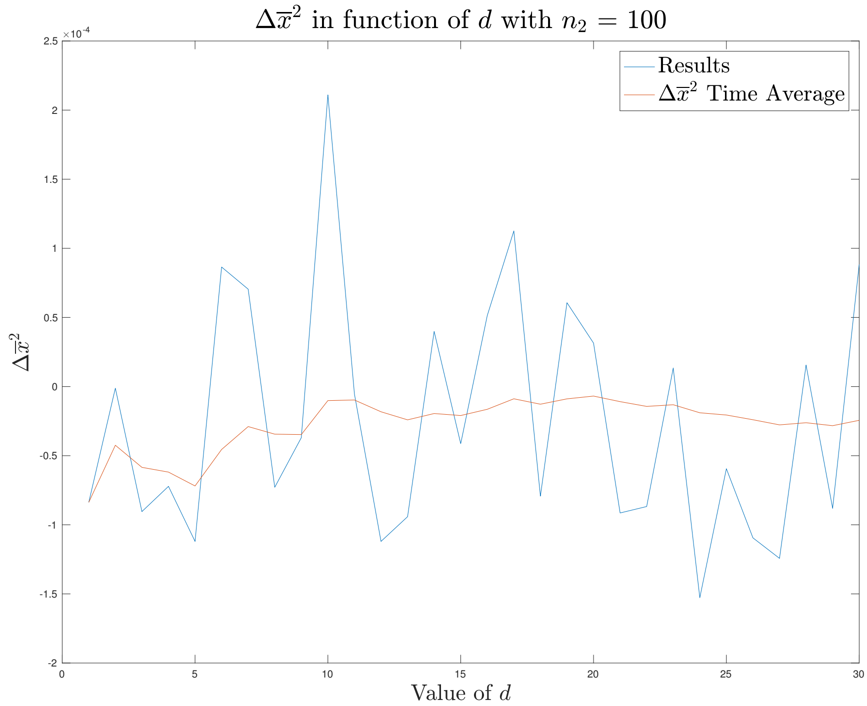


Figure C.1: In blue, the average gap on 10 simulations between the true square mean value (see Notation) for a barbell graph made of $n_2 = 100$ agents following d and the true square mean value of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

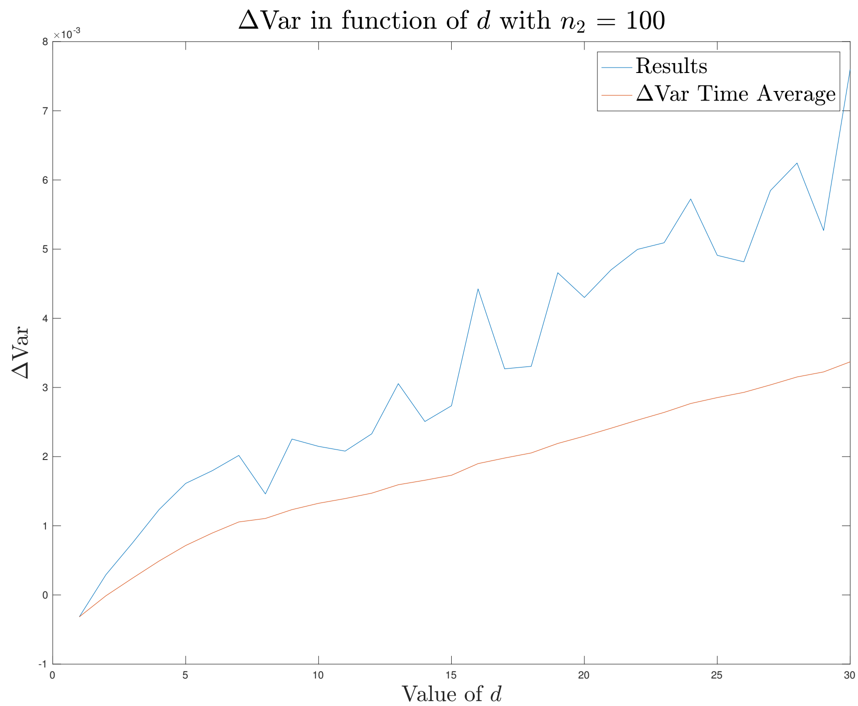


Figure C.2: In blue, the average gap on 10 simulations between the true variance (see Notation) for a barbell graph made of $n_2 = 100$ agents following d and the true variance of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

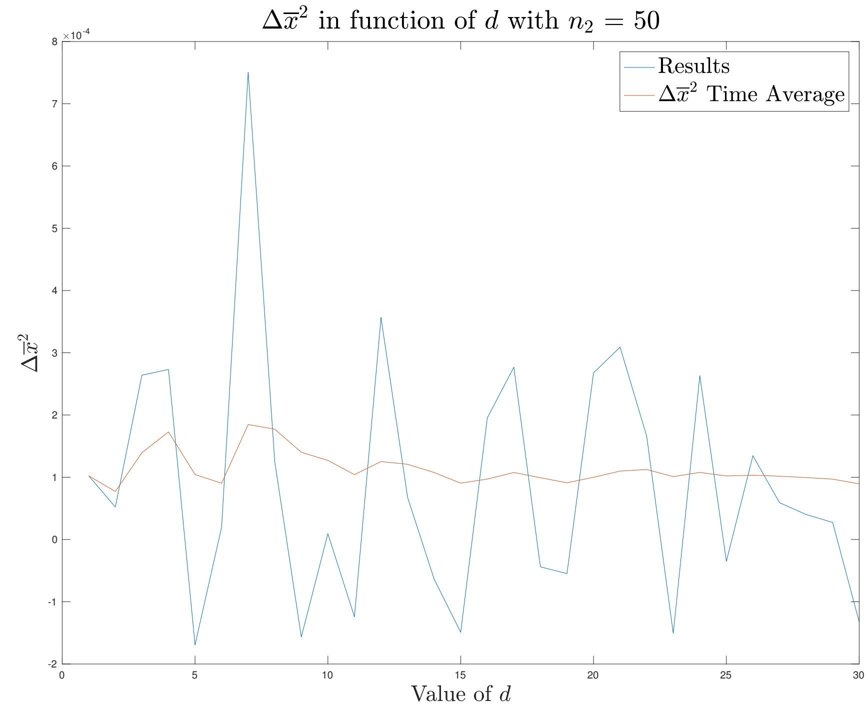


Figure C.3: In blue, the average gap on 10 simulations between the true square mean value (see Notation) for a barbell graph made of $n_2 = 100$ agents following d and the true square mean value of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

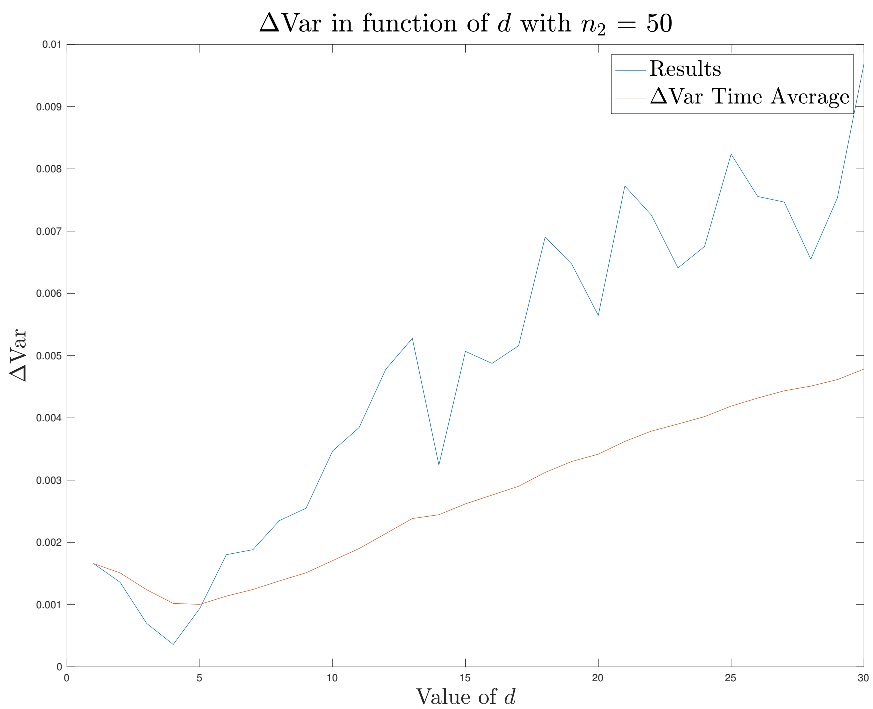


Figure C.4: In blue, the average gap on 10 simulations between the true variance (see Notation) for a barbell graph made of $n_2 = 50$ agents following d and the true variance of a complete graph with the same number of agents after 10000 time steps. In red, the time-average (Definition 1.1.2) of the blue curve.

Bibliography

[A.02] Deshmukh A. *KNOWLEDGE MANAGEMENT, ORGANIZATIONAL INTELLIGENCE AND LEARNING, AND COMPLEXITY - Vol. I - Distributed Artificial Intelligence*. 2002.

[AAR16] Jeremy Pitt Didac Busquet Alexander Artikis, Marek Sergot and Régis Riveret. Specifying and executing open multi-agent systems. August 17, 2016.

[BGPS06] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, June 2006. doi:10.1109/TIT.2006.874516.

[BHOT05] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Convergence in multiagent coordination, consensus, and flocking. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2996–3000, Dec 2005. doi:10.1109/CDC.2005.1582620.

[BHT09] V. D. Blondel, J. M. Hendrickx, and J. N. Tsitsiklis. On krause's multi-agent consensus model with state-dependent connectivity. *IEEE Transactions on Automatic Control*, 54(11):2586–2597, Nov 2009. doi:10.1109/TAC.2009.2031211.

[DAW12] J. C. Duchi, A. Agarwal, and M. J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic Control*, 57(3):592–606, March 2012. doi:10.1109/TAC.2011.2161027.

[dK] Maurice de Kunder. The size of the world wide web (the internet). http://www.worldwidewebsize.com. [Online; accessed 29-May-2018].

[FK15] Alan Frieze and Michal Karonski. *Introduction to Random Graphs*. Cambridge University Press, 2015.

[GJ12] Sébastian Grauwin and Pablo Jensen. Opinion group formation and dynamics: Structures that last from nonlasting entities. *Phys. Rev. E*, 85:066113, Jun 2012. URL: https://link.aps.org/doi/10.1103/PhysRevE.85.066113, doi:10.1103/PhysRevE.85.066113.

[Hew91] C. Hewitt. *Open information systems semantics for distributed artificial intelligence. Artificial Intelligence, 47:79–106.* Elsevier Science Publishers Ltd. Essex, UK, 1991.

[HM17] J. M. Hendrickx and S. Martin. Open multi-agent systems: Gossiping with random arrivals and departures. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 763–768, Dec 2017. doi:10.1109/CDC.2017.8263752.

[JL14] Jeffrey D. Ullman Jure Leskovec, Anand Rajaraman. *Mining of Massive Datasets*. Cambridge University Press, 2014.

[KBS06] A. Kashyap, T. Basar, and R. Srikant. Consensus with quantized information updates. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 2728–2733, Dec 2006. doi:10.1109/CDC.2006.376993.

[MCZ15] Tomaz Pisanski Marston Conder and Arjana Zitnik. Vertex-transitive graphs and their arc-types. May 6 2015.

[Mor05] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, Feb 2005. doi:10.1109/TAC.2004.841888.

[OSFM07] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, Jan 2007. doi:10.1109/JPROC.2006.887293.

[RFTI15] C. Ravazzi, P. Frasca, R. Tempo, and H. Ishii. Ergodic randomized algorithms and dynamics over networks. *IEEE Transactions on Control of Network Systems*, 2(1):78–87, March 2015. doi:10.1109/TCNS.2014.2367571.

[Rob16] Eric Roberts. The google pagerank algorithm. November 2016.

[Sal07] Jimmy Salvatore. Bipartite graphs and problem solving. August 8, 2007.

[SB98] Lawrence Page Sergey Brin. The anatomy of a large-scale hypertextual web search engine. 1998.

[Weia] Eric W. Weisstein. Graph automorphism. http://mathworld.wolfram.com/GraphAutomorphism.html. [Online; accessed 30-May-2018].

[Weib] Eric W. Weisstein. Graph diameter. http://mathworld.wolfram.com/GraphDiameter.html. [Online; accessed 21-May-2018].

[Wei99] Gerhard Weiss. *Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence.* The MIT Press Cambridge, Massachusetts London, England, 1999.

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve www.uclouvain.be/epl