

Algorithms in Open Multi-Agent Systems

Gossiping with Random Replacements

Dissertation presented by
Baptiste BRASSEUR

for obtaining the Master's degree in
Mathematical Engineering

Supervisor(s)
Julien HENDRICKX

Reader(s)
Raphaël JUNGERS, Anthony PAPAVALIOU

Academic year 2017-2018

Acknowledgements

First of all, I wish to address my sincere gratitude to my supervisor, Prof. Julien Hendrickx, for his precious advice that lead to the realization of this master thesis as well as for the support he provided during this past year.

I also want to thank my godfather and my father for the proofreading of my work.

Last but not least, I thank my family, friends and roommates for their patience and support.

Contents

Introduction	1
1 Description of the System	5
2 Presentation of the Basic Algorithms	7
2.1 Uncorrected Gossip Algorithm	7
2.2 Corrected Gossip or Last Message Algorithm	9
2.3 Conclusion	14
3 Numerical Comparison Between the Uncorrected Gossip Algorithm and the Last Message Algorithm	15
3.1 First Approach	15
3.1.1 3 Agents System	15
3.1.2 10.000 Agents System	18
3.2 Constant Distribution of the Incoming Values	20
3.2.1 Analysis of the Process	20
3.2.2 Analysis of the Asymptotic Errors	26
3.3 Time-Depending Distribution of the Incoming Values	32
3.3.1 Evolution of the Mean of the Values in the System	32
3.3.2 Uncorrected Message and Last Message Strategies	33
3.4 Conclusion	37
4 Reserve-Based Algorithm	39
4.1 Description of the Algorithm	39
4.2 Constant Distribution of the Incoming Values	42
4.2.1 Analysis of the Process: Influence of k_1	42

4.2.2	Analysis of the Asymptotic Errors: Influence of k_1	44
4.2.3	Analysis of the Process: Influence of k_2	46
4.2.4	Analysis of the Asymptotic Errors: Influence of k_2	48
4.3	Time-Depending Distribution of the Incoming Values	50
4.3.1	Influence of k_1 and k_2	51
4.4	Conclusion	52
5	Design of a New Strategy	55
5.1	Replacements in the Uncorrected Gossip Algorithm and in the Last Message Algorithm	56
5.2	Case 1: Message Sent to the Incoming Agent and Consensus Reached . . .	62
5.3	Case 2: Message Sent to the Incoming Agent and Consensus Not Reached .	66
5.4	Case 3: Message Sent to a Random Agent in the System	69
5.4.1	Alternative Strategy	69
5.4.2	Analysis of the Process	71
5.4.3	Analysis of the Asymptotic Error	73
5.5	Conclusion	74
	Conclusion	77
	Appendices	79
	A Additional Results	81
	B Notations and Symbols	85
	C Matlab Codes	87
	Bibliography	99

Introduction

Theory of coupled oscillators, system diagnosis, neural networks, economics, even the *rendezvous* problem [AB18], [XL17] or the flock of birds [HM17]; some fields that seem unrelated at first sight, and yet, each of them can be assimilated to a multi-agent system.

A multi-agent system is a loosely coupled network of problem-solving entities (agents) that work together to find answers to problems that are beyond the individual capabilities or knowledge of each entity [Gla06]. In this report, the author also defines an agent as anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors.

To illustrate those definitions, the *rendezvous* problem can be solved by modelling the problem as a multi-agent system where people (agents) communicate with each other in order to find a single unspecified location to meet [Ren04]. [RKJ16] proposes a multi-agent system for economic relations where the population (agents), that is attributed certain characteristics, interacts with the goal to exist.

The study, construction and application of multi-agent systems is called distributed (or decentralized) artificial intelligence (DAI), which is a subfield of artificial intelligence (AI). It appeared in the mid to late 1970s and is already an established and promising research and application field [Wei99]. This is among other things indicated by the fact that more industries are looking for ways to use multi-agent systems in order to use agent-based technologies [XL17]. Distributed AI refers to the fact that the information is shared between the agents and that this approach does not rely on a central controller to coordinate all the tasks. This leads to a first advantage, namely robustness, since the system may therefore tolerate the failure of one or more agents because other agents in the system may take over their part [A.02], [Wei99]. [A.02] also outlines that DAI offers the promise of higher computational speeds by using inherent asynchrony and parallelism in agents. Another advantage of this approach is its scalability. [Gla06] indeed states that it should be easier to add new agents to a multi-agent system than to add new capabilities to a monolithic system.

For [A.02], the two main reasons to cause people to study multi-agent systems are on the one hand the technological and applications needs and on the other hand the natural view of intelligent systems. The former relates the fact that such systems offer a promising and innovative way to deal with number of computing and information systems such as the Internet. It is also a way to provide solutions to problems that are hard to solve with centralized computing technology in the domains of scheduling, manufacturing, control and diagnosis for example. The latter is due to DAI's ability to provide insights and understanding about poorly understood interactions among natural, intelligent beings,

as they organize themselves into various groups, committees, societies, and economies in order to achieve improvement.

Literature does not usually consider multi-agent systems as being open, meaning that agents are able to join or leave the system while the studied process takes place. The composition and size of such systems may therefore evolve through time. While in closed systems, researchers are able to characterize the long term behavior of a multi-agent system such as convergence and synchronization [HM17], this task becomes much more complex when this assumption is not made.

This master thesis focuses on such open multi-agent systems. The problem that we are trying to solve is the average seeking through gossip. In other words, the agents join the system with a value of their own and their common goal is to approximate as accurately as possible the average of the values in the system. In order to achieve this objective, they interact through pairwise gossips. It means that they draw information concerning their environment and update their approximation by communicating in pairs with other agents. We assume throughout this report that each agent in the system may interact with any of its peers also present in the system. That is, if we look at a system as a graph where the nodes are the agents present in the system and an edge exists between two of them allowing them to gossip, the complete graph is considered.

In open systems, agents are allowed to join or leave the system. We will however restrict the problem to the case where leaving agents are immediately replaced. Hence, we will focus on systems whose size remains unchanged as the processes take place.

This master thesis starts from two naive, basic algorithms. Its first objective will be to characterize them as accurately as possible. We will study their theoretical behavior. We will also analyze their numerical results throughout time as well as their asymptotic characteristics. This will lead us to a first comparison based on the advantages and drawbacks of those two strategies. We will therefore try to take the best of each of them in order to find algorithms that lead to smaller errors than those first two. The final objective of this work is to determine which of those strategies is the most adapted according to the parameters, i.e. the gossip rate and the size of the system.

In Chapter 1, we will present the systems that will be considered throughout this master thesis as well as the assumptions that will be made. We will also present more accurately the problem that those algorithms are meant to solve in this work.

Chapter 2 is dedicated to an initial set of two algorithms designed to solve the problem of consensus seeking systems. On the one hand there is a naive strategy, i.e. the uncorrected gossip strategy. This strategy is linked to a source of error coming from the replacement steps. Indeed, the agents do not announce their departure as they leave the system. An error (that we will define as group error) is therefore inserted in the system. On the other hand, a second algorithm, the last message strategy, is then designed in order to make up for that unwanted behavior. In fact, the leaving agents send a message to an agent in the system that rectifies its approximation in order to avoid that so-called group error. The main theoretical characteristics of these algorithms are presented in that chapter and are illustrated by basic numerical results.

Chapter 3 is devoted to the analysis of the numerical results. We will first present the evolution of a small system during a short process in order to have a first contact with

those strategies and to build a first intuition about the upcoming. On that basis, the dependence of the two methods on the different parameters describing the systems will be proposed. We will first observe the behavior of both strategies during processes before analyzing their asymptotic characteristics. We will find that the messages may be a source of error. And yet we still want to avoid group error.

In Chapter 4, we will therefore design a strategy that can be qualified as "algorithm with reserve". In that strategy, the messages sent by the leaving agents are not directly included to the approximation of the agents in the system that receive them. They store them instead as *reserve* or *surplus* that are eventually reincluded into the approximations of the agents during gossip steps. The hope is that this strategy may on the one hand avoid the group error and on the other hand avoid the error linked with the last message. Actually the reserves are an upgrade of the last message strategy but do not provide the expected results, which leads to the content of the final chapter.

In Chapter 5, we will finally design a strategy that tries to take the best of each of the three methods considered so far. Indeed, the aim of this strategy is to keep a zero-group error as in the last message strategy or the reserve-based algorithm while keeping a small error due to the last message as in the uncorrected gossip strategy. Therefore, this new *alternative* strategy will be close to the algorithm with reserve. The difference is that the agents to whom the message has been sent are making a decision on whether to store it entirely or partly as reserve or to include it to their own approximation. This decision is made according to the value of the message and the approximations of both interacting agents.

Finally, please note that in each chapter, two main scenarios will be discussed. The distribution of values held by the agents joining the system will be either constant or evolving through time.

Chapter 1

Description of the System

Before presenting the first two strategies that will be analyzed, let us give a description of the system that will be studied in this master thesis. In this first chapter, we also introduce the main assumptions that are made in the upcoming results.

In this work, we will consider multi-agent systems where the interactions between agents are pairwise gossips. Those systems are assumed to be open (agents are allowed to join or leave the system) so that their composition is evolving with time. We denote by $\mathcal{S}(t)$ the set of agents present in the system at time t . Integers are used to designate agents such that $\mathcal{S}(t) \subset \mathbb{N}$. In the rest of this work, we will consider replacements in such a way that the number n of agents in a system is fixed.

All the agents in the system are assigned two values:

- a value of their own r_i . This value is supposed to remain constant for an agent during its time in the system;
- an approximation $x_i(t)$ that is made about the average of the system. This value may evolve with time (through gossips for example).

We assume the value of each agent to follow a normal distribution $\mathcal{N}(\mu(t), \sigma^2)$. For each strategy, we will firstly consider a constant value for the mean of the distribution. This value will be arbitrary set to 0. Secondly, $\mu(t)$ will be assumed to depend on time. In that scenario, it will increase linearly between 0 and a certain value μ_{max} over t_{max} time steps. The variance of the distribution is assumed to remain constant for each process and is chosen to be unitary.

The objective of the interactions is to lead each agent in the system to have an approximation of the average of the values in the system. Hence the error that is considered in this work is the Mean Square Error (MSE) defined as

$$\mathbb{E}[(\mathbf{x} - \bar{r})^2]$$

where $\mathbf{x}$ represents the set of the approximations in the system and $\bar{r}$ represents the average of the values in this same system at the corresponding time. Note that the time dependence of those values is neglected here to lighten the expression.

Finally, we assume a discrete evolution of time $t \in \mathbb{N}$ where each time step corresponds to an instant of the continuous time process where an event occurred (gossip or replacement). We define as p_{gos} the probability that a gossip occurs for any time step, and therefore $p_r = 1 - p_{gos}$ is the probability that the event is a replacement.

Chapter 2

Presentation of the Basic Algorithms

The aim of this chapter is to present the two main strategies that will interest us and that are the basis for this master thesis. We will on the one hand introduce the uncorrected gossip algorithm whose replacement steps are not associated with any interactions. On the other hand, we will present the last message (or corrected gossip) algorithm. In this last strategy, the leaving agents announce their departure and broadcast a last message to a random agent in the system. Those two strategies are meant to solve the problem where the interactions between agents lead to pairwise gossip averages.

2.1 Uncorrected Gossip Algorithm

The first strategy that is presented is the uncorrected gossip algorithm [HM17]. In this strategy, somehow basic, the two types of events that may occur (gossip or replacement) are defined as follows.

Gossip: Two agents $i, j \in \mathcal{S}(t)$ are uniformly randomly and independently selected among the n agents present in the system¹, and they update their approximations by performing a pairwise average:

$$x_i(t+1) = x_j(t+1) = \frac{x_i(t) + x_j(t)}{2}.$$

■

Algorithm 1 Uncorrected Gossip Algorithm: Gossip Step

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t)$

Select uniformly and randomly two agents $i, j \in \mathcal{S}(t)$

$$x_i(t+1) = x_j(t+1) = \frac{x_i(t) + x_j(t)}{2}$$

Replacement: One uniformly randomly selected agent $l \in \mathcal{S}(t)$ leaves the system and is replaced by a "new" agent $e \notin \mathcal{S}(s), \forall s \leq t$, that joins the system. Therefore $\mathcal{S}(t+1) \neq \mathcal{S}(t)$ but $n(t+1) = n(t) = n$. ■

¹In this master thesis, we suppose that an agent cannot gossip with itself as it is accepted in paper [HM17].

Algorithm 2 Uncorrected Gossip Algorithm: Replacement Step

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t)$

Select uniformly and randomly an agent $l \in \mathcal{S}(t)$

New agent $e \notin \mathcal{S}(s), \forall s \leq t$ ($x_e(t+1) = r_e$)

$\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

In other words during a gossip step, the two interacting agents update their approximation with the mean of both of theirs. In a replacement step, an agent leaves the system without announcing its departure and another one joins it without any further implications (hence *uncorrected* gossip algorithm). The latter takes as a first approximation its own value since it has initially no information on the mean of the system. The other agents in the system (that are not included in an interaction) keep their approximation unchanged.

Given the definition of the replacement, a loss of information and accuracy occurs in the system. Let us prove that the replacement step may imply $\bar{x}(t)$ to be different from $\bar{r}(t)$. In the proof below, we will use " $\neq$ " to mean "not equal for sure".

Proof: Let us prove that $\frac{1}{n} \sum_{i \in \mathcal{S}(t)} x_i(t) \neq \frac{1}{n} \sum_{i \in \mathcal{S}(t)} r_i$ after a replacement. This event is composed by two combined phenomena:

- an agent l leaves the system,
- a "new" agent e comes in the system, holding a new value and an approximation of the average of the system such that $x_e(t+1) = r_e$.

Let us suppose that, for a given time t , $\bar{x}(t) = \bar{r}(t)$.

First, let us show that the entry of an agent e does not imbalance this equation for time $t+1$. Indeed,

$$\begin{aligned} \frac{1}{n(t+1)} \sum_{i \in \mathcal{S}(t+1)} x_i(t+1) &= \frac{1}{n(t+1)} \left(\sum_{i \in \mathcal{S}(t)} x_i(t) + x_e(t) \right) \\ &= \frac{1}{n(t+1)} \left(\sum_{i \in \mathcal{S}(t)} r_i + r_e \right) \\ &= \frac{1}{n(t+1)} \sum_{i \in \mathcal{S}(t+1)} r_i \end{aligned}$$

Let us therefore show that the error is linked to the leaving agent l .

$$\begin{aligned} \frac{1}{n(t+1)} \sum_{i \in \mathcal{S}(t+1)} x_i(t+1) &= \frac{1}{n(t+1)} \left(\sum_{i \in \mathcal{S}(t)} x_i(t) - x_l(t) \right) \\ &\neq \frac{1}{n(t+1)} \left(\sum_{i \in \mathcal{S}(t)} r_i - r_e \right) \\ &= \frac{1}{n(t+1)} \sum_{i \in \mathcal{S}(t+1)} r_i \end{aligned}$$

The inequality arises from the fact that, if an agent gossiped during its time in the system, it may have changed its approximation (hence $r_l \neq x_l(t)$). ■

We can show that there are two terms of error linked with this algorithm. An obvious disagreement term results from the variance of the distribution. The gossip steps in that strategy are therefore meant to decrease that error. The second error that we have just outlined is a group error coming from the agents leaving the system without any warning.

Let us rewrite the expression of the MSE as follows:

$$MSE = \mathbb{E}[(\mathbf{x} - \bar{r})^2] = \mathbb{E}[(\mathbf{x} - \bar{x}) + (\bar{x} - \bar{r})]^2 \approx \mathbb{E}[(\mathbf{x} - \bar{x})^2] + \mathbb{E}[(\bar{x} - \bar{r})^2].$$

The last approximation comes by assuming independence between $(\mathbf{x} - \bar{x})$ and $(\bar{x} - \bar{r})$ and $\mathbb{E}[(\mathbf{x} - \bar{x})] = 0$.

We have therefore addressed the two terms of error associated with the uncorrected gossip strategy:

- the disagreement between the agents $\mathbb{E}[(\mathbf{x} - \bar{x})^2]$,
- the group error $\mathbb{E}[(\bar{x} - \bar{r})^2]$.

Figure 2.1 represents the evolution of the three approximations of a system where the agents are follow the uncorrected gossip algorithm. For example, at time step 1, agent 2 (red dot) gossips with agent 3 (yellow dot) that leads them to the same approximation at time step 2. While the approximations seem to converge initially to the true average of the system, this is not the case anymore when a replacement occurs (time step 8).

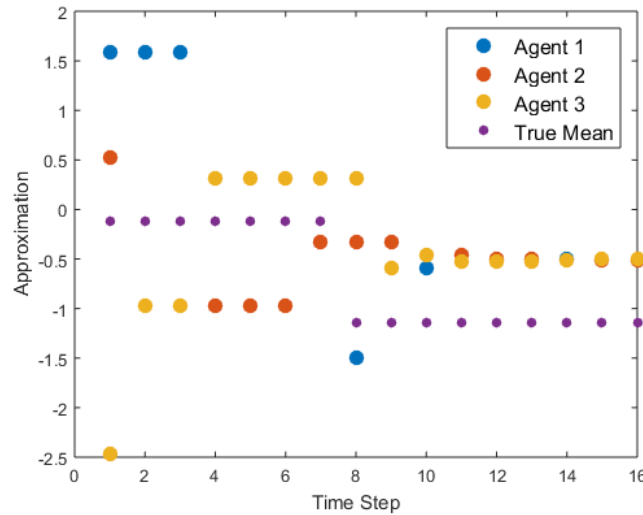


Figure 2.1: Illustration of the evolution of the three approximations in the system considering the uncorrected gossip algorithm.

2.2 Corrected Gossip or Last Message Algorithm

As a natural consequence, we desire to design an algorithm that avoid the group error introduced in the previous section by adapting the replacement step. In other words we

would like to find a strategy such that $\bar{r}(t) = \bar{x}(t)$ at each time of the process. This is feasible by forcing the leaving agent to send a last, well chosen, message to an agent in the system. The gossip step is the same as in the uncorrected gossip strategy presented above. The replacement steps are treated as follows.

Replacement: Two uniformly randomly agents $l, m \in \mathcal{S}(t)$ are selected. The leaving agent l computes its Δ - value, that is $\Delta_l(t) = x_l(t) - r_l$, and sends it to the agent m whose approximation is updated as

$$x_m(t+1) = x_m(t) + \Delta_l(t).$$

The agent l then leaves the system and is replaced by a "new" agent $e \notin \mathcal{S}(s)$, $\forall s \leq t$, that joins the system. Therefore $\mathcal{S}(t+1) \neq \mathcal{S}(t)$ but $n(t+1) = n(t) = n$. ■

Algorithm 3 Last Message Algorithm: Replacement Step

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t)$

Select uniformly and randomly two agents $l, m \in \mathcal{S}(t)$

$\Delta_l(t) = x_l(t) - r_l$

$x_m(t+1) = x_m(t) + \Delta_l(t)$

New agent $e \notin \mathcal{S}(s)$, $\forall s \leq t$ ($x_e(t+1) = r_e$)

$\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

In other words, this strategy deals with the gossip in the same way as the uncorrected gossip algorithm does but the leaving agent sends a message to an agent in the system when it leaves it. This message is the variation of the approximation undergone by the leaving agent during its time in the system.

Let us now prove that this algorithm has indeed the expected characteristics.

Proof: Let us prove that $\frac{1}{n} \sum_{i \in \mathcal{S}(t)} x_i(t) = \frac{1}{n} \sum_{i \in \mathcal{S}(t)} r_i$ at each time of the process. This proof will be made by recursion.

Initialization: Each agent enters the system with an approximation equal to their own value ($x_i(t_0) = r_i$, $\forall i \in \mathcal{S}(t_0)$). The statement is therefore initially true.

Recursion Assumption: Let us suppose the proposition is true for a given time $t > t_0$. Let us show that it is also true at time $t+1$.

Recursion Step: Three different events may occur:

- the agents i, j gossip
- the agent l leaves the system and broadcasts a message to agent m
- the agent e enters the system

Keep in mind that the last two scenarios combined form a replacement. Let us show that each of these steps respects the proposition.

- Gossip:

$$\begin{aligned}
\frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} x_k(t+1) &= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} x_k(t) - x_i(t) - x_j(t) + 2 \cdot \frac{x_i(t) + x_j(t)}{2} \right) \\
&= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} x_k(t) \right) \\
&= \frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} r_k
\end{aligned}$$

The last equation comes from noticing that $\mathcal{S}(t+1) = \mathcal{S}(t)$ and by using the recursion hypothesis.

- Arrival:

$$\begin{aligned}
\frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} x_k(t+1) &= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} x_k(t) + x_e(t) \right) \\
&= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} r_k + r_e \right) \\
&= \frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} r_k
\end{aligned}$$

The first and last equalities come from $\mathcal{S}(t+1) = \mathcal{S}(t) \cup \{e\}$. The second one is derived from the recursion hypotheses and $x_e(t) = r_e$.

- Departure:

$$\begin{aligned}
\frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} x_k(t+1) &= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} x_k(t) - x_l(t) - x_m + (x_m + \Delta_l(t)) \right) \\
&= \frac{1}{n(t+1)} \left(\sum_{k \in \mathcal{S}(t)} r_k - x_l(t) - x_m(t) + (x_m + x_l(t) - r_l) \right) \\
&= \frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t)} r_k - r_l \\
&= \frac{1}{n(t+1)} \sum_{k \in \mathcal{S}(t+1)} r_k
\end{aligned}$$

The first equality comes from $\mathcal{S}(t+1) = \mathcal{S}(t) \setminus \{l\}$ and from the last message sent to a random agent of the system. The second one is derived from the recursion hypotheses and the value of the last message. The last one comes again from the update of the set of agents.

We can therefore conclude that $\bar{r}(t) = \bar{x}(t)$ at each time of the process. ■

We have just shown that we found a strategy that enables the agents to gossip in order to find the *true* mean of the system, which was not the case in the uncorrected gossip strategy. This result is shown on figure 2.2. This figure represent a similar situation as in previous section but for a different system.

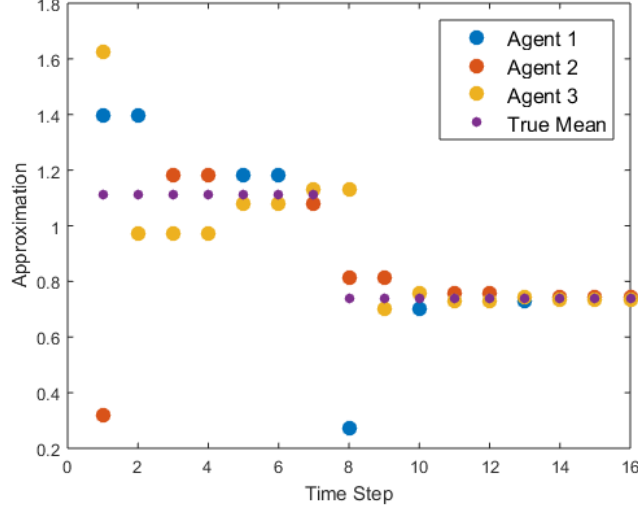


Figure 2.2: Illustration of the evolution of the three approximations in the system considering the last message algorithm.

Let us now show that this strategy also has the following property: suppose that the system closes at a given time t_c (agents are not able to enter or leave the system and the agents may therefore only gossip), then the approximation of each agent converges to the true mean of the system so that $x_i(t) \rightarrow \bar{r}(t) = \bar{r}(t_c)$, $\forall i \in \mathcal{S}(t_c)$ as $t \rightarrow \infty$.

Proof: This proof will be achieved in two steps. First we will prove that each agent's approximation does indeed converge toward a certain value. We will then show that this value is the true mean of the system $\bar{r}(t_c)$.

The convergence is proved with the following algorithm [VDBT10]:

THEOREM 1: Consider the discrete-time consensus seeking systems of the following kind: each agent i starts with a scalar value $x_i(0)$. The vector $x(t) = (x_1(t), \dots, x_n(t))$ with the values held by the agents at time t , is updated according to the equation $x(t+1) = A(t)x(t)$, or

$$x_i(t+1) = \sum_{j=1}^n a_{ij}x_j(t),$$

where $A(t)$ is a non-negative matrix with entries $a_{ij}(t)$, and where the updates are carried out at some discrete set of times.

The communication pattern at each step can be described in terms of a directed graph $G(t) = (N, E(t))$, where $(i, j) \in E(t)$ if and only if $a_{ij} > 0$.

Suppose that

1. There exists a positive constant α such that
 - $a_{ii}(t) \geq \alpha$, for all i, t ,
 - $a_{ij}(t) \in \{0\} \cup [\alpha, 1]$ for all i, j, t ,
 - $\sum_{j=1}^n a_{ij}(t) = 1$, for all i, t .
2. Graph $(N, \cup_{s \geq t} E(s))$ is strongly connected for all $t \geq 0$.

3. If i communicates to j an infinite number of times (that is, if $(i, j) \in E(t)$ infinitely often, then there is some bound B such that, for all t , $(i, j) \in E(t) \cup E(t+1) \cup \dots \cup E(t+B-1)$.

Under previous assumptions, the agreement algorithm guarantees asymptotic consensus, that is there exists some c such that $\lim_{t \rightarrow \infty} x_i(t) = c$, for all i . $\square$

We can apply this theorem to our problem. Indeed, when the system closes, each agent holds an approximation $x_i(t_0)$ and for each time $t \geq t_0$, they update their value as

$$x_i(t+1) = \begin{cases} \frac{x_i(t) + x_j(t)}{2} & \text{if agents } i, j \text{ gossip,} \\ x_i(t) & \text{otherwise.} \end{cases}$$

Therefore $a_{ii}(t) = a_{jj}(t) = a_{ij}(t) = a_{ji}(t) = \frac{1}{2}$ if agents i, j gossip at time t . For all the other agents, $a_{kk}(t) = 1$. The other elements of $A(t)$ are null.

The first assumption is respected considering $\alpha = \frac{1}{2}$. The connectivity of the graph is assured by the definition of the problem. Finally, the bounded intercommunication intervals are respected if each pair of agents i, j interacts at least each B time steps which we can consider as true (B obviously grows with the number of agents in the system).

We can conclude this proof by showing that $c = \bar{r}(t_0)$. Indeed, since each agents in the system asymptotically holds an approximation equal to c , using $\bar{r}(t) = \bar{x}(t)$, it comes

$$\bar{r}(t) = \bar{x}(t) = \frac{1}{n} n \cdot c = c$$

■

Figure 2.3 outlines this last property for a system composed of 1000 agents that closes after 7500 time steps.

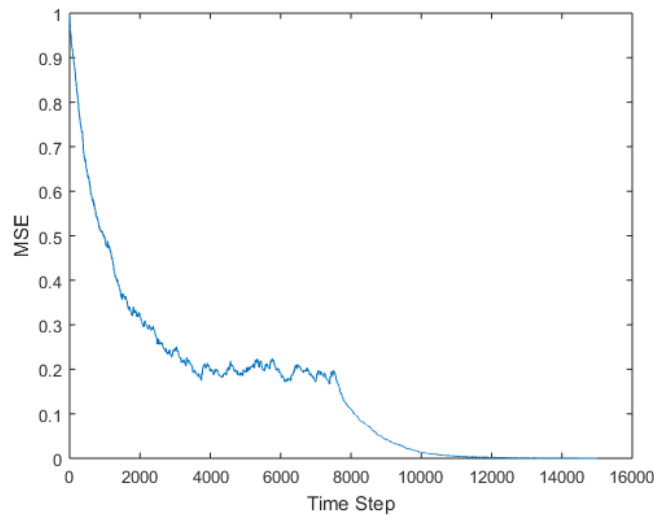


Figure 2.3: Illustration of the convergence toward zero of the mean square error of the last message strategy after the closure of the system.

2.3 Conclusion

In this second chapter, we presented two methods to solve the problem of average seeking systems. On the one hand, we introduced a naive strategy that is the uncorrected gossip strategy. The agents in the system update their approximation of the average of the system through symmetric pairwise gossips. The replacement steps are not linked with any interaction, which is a source of error. Given that the leaving agents in the system do not announce their departure, a group error is inserted in the system.

We therefore consider another algorithm in which the leaving agent broadcasts a last message to an agent in the system. This strategy is meant to suppress the group error. Therefore, we have shown that the agents are able to reach an asymptotic consensus on the mean of the values present in a system after its closure.

We will see in next chapter that, even if the last message strategy provides for two valuable properties, this strategy has to be taken carefully. It may even lead to unexpected (and unwanted) results, than may be less accurate than those coming from the uncorrected gossip algorithm.

Chapter 3

Numerical Comparison Between the Uncorrected Gossip Algorithm and the Last Message Algorithm

Let us now analyze the numerical results associated to both algorithms introduced in Chapter 2. Throughout this chapter, these methods will be compared. It will help to outline the advantages and drawbacks of both strategies coming from the strengths or weaknesses of one strategy against the other one. Keep in mind that they both depend on two parameters that are the gossip rate and the size of the system. To introduce the numerical results and in order to get familiar with those algorithms, we will first consider the simple case where the system is composed of only three agents. We will then extend this basic scenario to larger systems and analyze the influence of the gossip rate. The gossip rate is the probability, for a given time step, to observe a gossip rather than a replacement. We will first consider the case where the distribution of the incoming values is constant before tackling the problem where its mean increases as the process takes place.

3.1 First Approach

3.1.1 3 Agents System

Closed System

In order to have a first intuition about the results of both methods, let us consider the simple scenario where the system is composed of only three agents. We can start by observing the effect of the gossip steps on such a small system. The consensus seeking in the closed system is shown on figure 3.1 where each dot represents the approximation of an agent at a given time step. We can observe that the agents are able to find an accurate approximation of the mean of the system in only a few steps. We can easily extrapolate this scenario to a period between two replacements in an open system where the approximations will get closer to one another the more they interact. If they are able to gossip enough, they will reach an agreement, whether a good one (last message strategy)

or not (uncorrected gossip strategy).

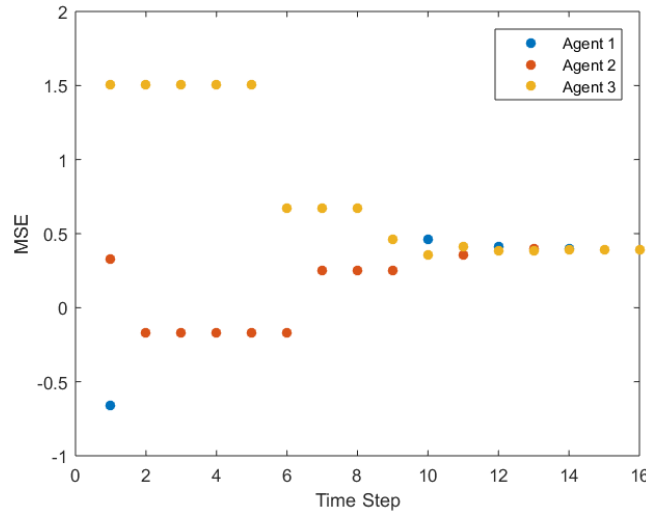


Figure 3.1: Evolution of the approximations of three agents in a closed system.

On that figure, we also see that the same pair of agents is allowed to gossip for successive time steps (i.e. time steps 1 to 4). The fact that the same two agents gossip several times in a row is obviously an unwanted phenomenon. Indeed, from the definition of *gossip*, used in those two algorithms (given in the previous chapter), a new gossip between two agents is useful (in the sense that they change their approximation) if at least one of them has updated its own approximation since their last interaction. Otherwise, the gossip has the same outcome over and over again. To illustrate that undesired phenomenon, among the fifteen gossips happening during the process represented on figure 3.1, only nine lead to an evolution of the state of the system. This tends of course to occur less frequently for large systems where randomly drawing twice the same pair of agents become unlikely.

Open System

Let us now introduce replacements in the process and observe their effect in both algorithms. Those results are represented on figure 3.2 for the uncorrected gossip strategy and on figure 3.3 for the last message algorithm (the same events occur in both representations). We still consider a system composed of three agents. On both figures the dots represent again the approximations of the three agents. If an agent gets replaced, so is its approximation with the value brought in the system by the new agent. The average of the values in the system is represented by the purple dots. This average is, of course, only evolving when replacements occur. The probability of observing a gossip for each time step is chosen to be equal to .9. With such a gossip rate and considering 3 agents in the system, the agents are likely to approach a consensus between two replacements.

We can observe on those figures two characteristics that were mentioned in Chapter 2: the presence of a group error for the uncorrected gossip algorithm and the convergence toward a consensus when enough gossips occur between two replacements. The first statement is particularly well illustrated here. Between time steps 20 and 40, the agents gossip until they reach a consensus at a value around .2 while the mean of the system is approximately

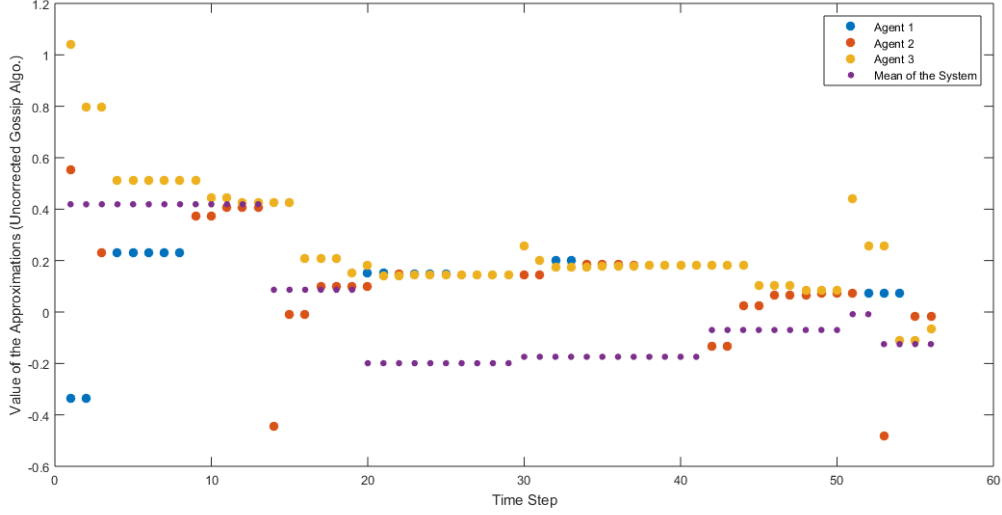


Figure 3.2: Evolution of the approximations of three agents in an open system using the uncorrected gossip strategy.

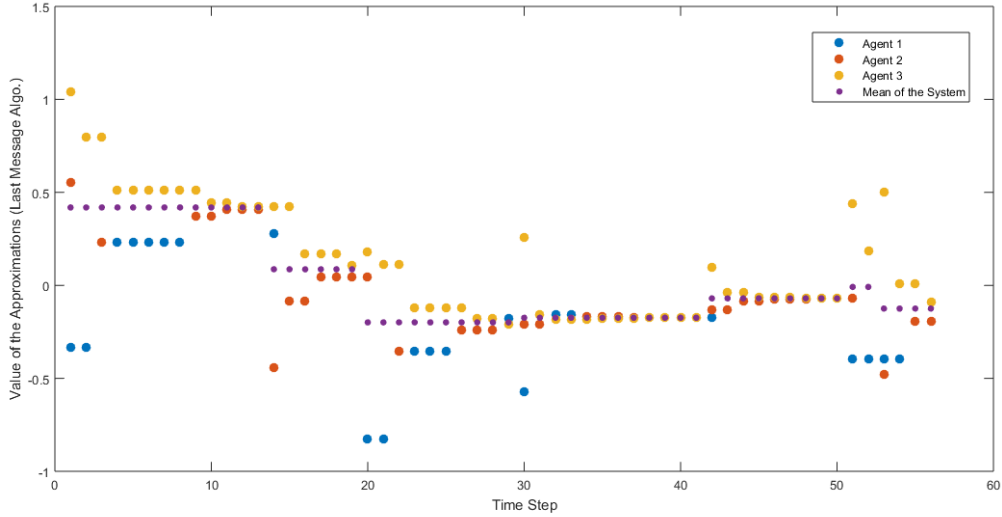


Figure 3.3: Evolution of the approximations of three agents in an open system using the last message strategy.

-0.2. Using the definition of the group error given in previous chapter, this term is worth around .16. Given that the agents reach quickly an agreement for this type of small system (the disagreement quickly decreases), this error is significant.

Another behavior that has to be kept in mind, even if it may seem obvious, is that a replacement in the last message strategy is linked with the evolution of two approximations in the system: one directly coming from the replaced agent (x_l replaced by $x_e = r_e$) and one is due to the last message ($x'_m = x_m + \Delta_l$). Indeed, we consider that the agents joining the system may not receive the messages sent by the leaving agents. On the other hand, only one value is changed after a replacement in the uncorrected gossip strategy. This is an important phenomenon since we will see that this often causes a sudden raise of the

error in the last message strategy.

The evolution of the process is also represented on figure 3.4 and is interpreted in the form of the Mean Square Error (MSE). In particular, we can see that the error often increases to a large extent in the results of the last message algorithm. In such a small system, it does not seem to be a problem since the MSE usually drops importantly in a few time steps. Given the large gossip rate, thanks to the last message, the agents spend most of their time in the system making an error close to zero.

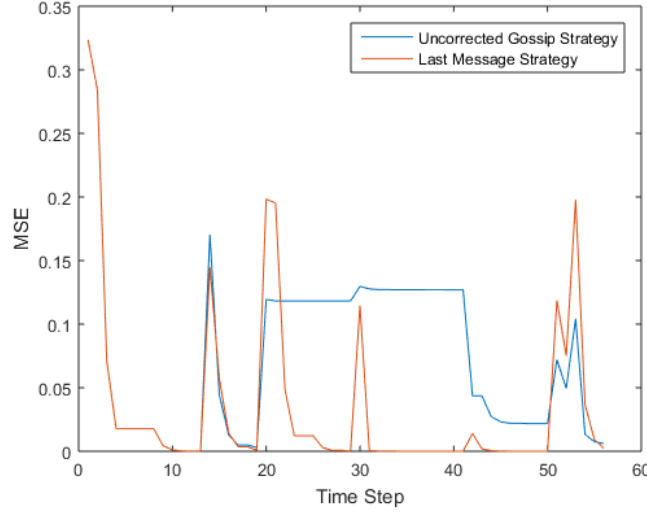


Figure 3.4: Mean Square Error associated to for the last message and the uncorrected gossip algorithm applied to the system represented on figure 3.2 and 3.3

The results of the uncorrected gossip strategy are different. The disagreement also rapidly drops thanks to the gossip steps that strongly decrease it in small systems (please note that, if the system is composed of 2 agents, an agreement is found in exactly one step). However, the agents may be gossiping to find a bad consensus which is represented for time steps 20 to 40. Also, on the contrary with the last message strategy, a replacement in the uncorrected gossip algorithm may be beneficial (see replacement at time steps 41). This comes from the fact that the lack of last message in that algorithm is obviously at the base of the group error that may occur with a replacement. Paradoxically, it also explains the possible decrease of the group error. Indeed, this error relates to the sum of the Δ – values that have left the system since the beginning of the process. Therefore, a new replacement may be increasing the group error if the agent is linked with a Δ_i that increases the magnitude of this sum, and decreases it otherwise.

3.1.2 10.000 Agents System

We have just observed the behavior of a system composed of three agents. Let us now consider a large one, whose size is 10.000 agents for example. It is obvious that the logic used in the beginning of this chapter cannot be extended to such system. Rather than considering each approximation and observe their evolution, we will observe the errors associated to each methods as the process takes place. In order to do so, we will analyze a mean process. In other words, given an initial state of the system and fixed parameters,

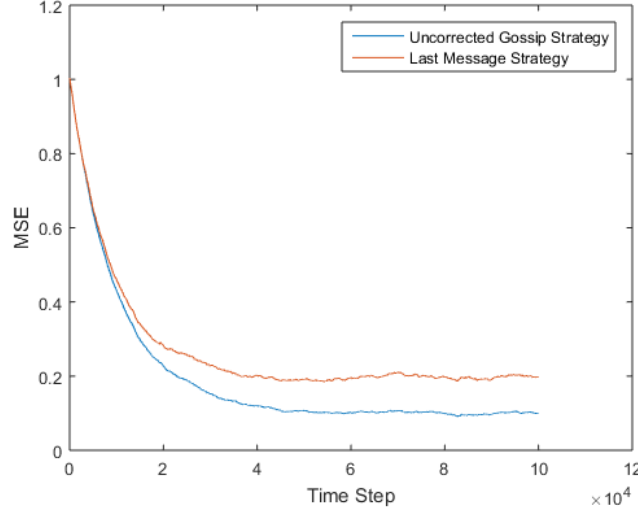


Figure 3.5: System composed of 10.000 agents for a $p_{gos} = .9$

the process is repeated a large number of times so that we can observe the mean evolution of the system. The results presented in the rest of this master thesis are coming, if not mentioned, from averaged processes.

First of all, for such large systems and given the normal distribution of the values joining them, the average of the system is supposed to be close to zero (law of large numbers). Also, each replacement has a small influence on the mean of the system. We can therefore assume that the group error for such large systems is small since the error introduced by a leaving agent is on the one hand weakly influencing the group error. On the other hand it is likely to be compensated by a later leaving agent (or to compensate the group error already made). Moreover, we can remember the comment made in the previous subsection about the modification of only one value in the system as a replacement occurs (replacement is usually a source of lower disagreement in the uncorrected gossip strategy than in the last message strategy). In large systems, it requires obviously a lot more gossips to reduce this disagreement than in small ones. This is indeed presented on figure 3.5.

In this first section, we had a first approach of both strategies considering small and large systems. We had the opportunity to get familiar with them and we had a first intuition about their practical results. We have seen that the last message strategy, even if it seemed to offer interesting theoretical characteristics, does not give as good results as expected. Although it seems to be appropriate to deal with small systems where it may be useful to take into account the group error, the uncorrected gossip strategy seems better adapted to deal with large ones. Indeed, we observed that the replacements, that may be a source of increase of the disagreement, can be compensated by quickly reaching an agreement thanks to the small size of the system in the last message strategy. On the other hand, the uncorrected gossip algorithm seems to be well adapted to large systems since the group error is expected to get close to zero, whatever the probability of gossip. Associated to the lower disagreement of this method (due to the evolution of only one approximation after a replacement step), this explains the good behavior of the uncorrected gossip algorithm that seems to stand out in such large systems.

3.2 Constant Distribution of the Incoming Values

It is now time to tackle the general problem and to present a more formal analysis of the two methods. In this section, we consider a constant distribution for the incoming values. Therefore, agents join the system with a value that is given according to a standardized normal distribution. We will first describe the behavior of both strategies as a function of time for several gossip rates and different systems. We will then analyze their asymptotic behavior depending on the different parameters.

3.2.1 Analysis of the Process

Let us begin by analyzing the process flow. In this part, we will try to outline the characteristics through time of both methods considering several values for the two parameters. Keep in mind that we will observe the error of a mean process.

Agents Dependency

Let us start with the influence of the size of the system. We will here fix the gossip rate and consider increasing sizes of the system from which we will draw the conclusion. 200 processes are simulated for each considered system and their mean is represented on the following graphs.

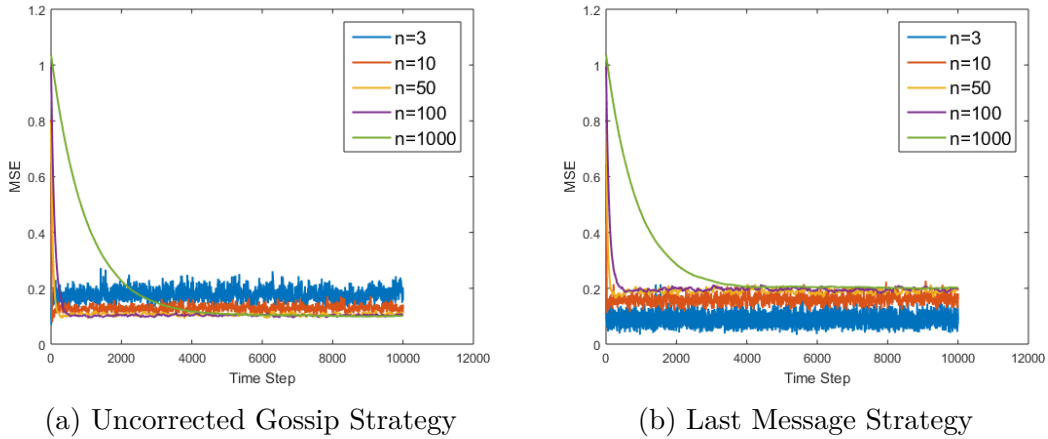


Figure 3.6: Mean evolution through time of the MSE for a probability of gossip of .9 for several sizes of systems.

Let us start the analysis considering figure 3.6 that represents the mean process for increasing sizes of systems given a gossip rate of .9. The first observation that jumps out in both algorithms is that the smoothness of the curves on those graphs strongly depends on the number of agents in the system. The larger the system, the smoother the corresponding curve. This is due to the fact that each event has a lower influence on the general error in large systems. Indeed, as we compute the mean square error, the error is weighted by the number of agents in the system. On the one hand, if a gossip step occurs, the system is subject to a larger decrease of the MSE, if it is composed of few agents. On the other hand, if an agent gets replaced, both the mean of the system and

the associated error are more impacted in small systems. It may therefore be useful to increase the number of repetitions of processes in order to smooth results.

Another remark that has already been implied and that is true for both strategies is that the mean process reaches an asymptotic error. While this is clear for large systems, it is less obvious for small ones. Indeed, from the large deviation between the results outlined above, the error seems to oscillate around a certain value. Increasing the number of repetitions leads to improved smoothness of the curves and therefore the same characteristic can be observed.

In both strategies, the larger the system, the longer¹ it needs for the error to reach its asymptotic value. Obviously, more gossips are required to include a similar proportion of the agents in the system to gossip if their number increases. Picture for example a system composed with 2 or 1000 agents. While the consensus can be reached in exactly one interaction in the first case, in the second one, a gossip is linked to an insignificant decrease of the disagreement.

Now that we have outlined the similarities of the two graphs, let us state their differences. The first one comes actually from a common point: in both strategies, the asymptotic values depend on the size of the system. However the uncorrected gossip strategy leads to a smaller asymptotic error as the size of the system increases, in contrast to the last message algorithm that is associated with opposite results. The asymptotic error that decreases with the number of agents comes from the fact that the group error may become really large for small systems and is nearly nonexistent in large ones. On the other hand, in the last message strategy, while the benefit of the agreement reached between two replacements compensates the large disagreement implied by the replacement in small systems, the long time required for agents to be included in a gossip makes the last message strategy poor for large systems.

Let us come back for a moment to the small group error linked with large systems. This may seem insignificant and natural and yet it appears to be an unexpected and important strength of the uncorrected gossip strategy as we see throughout this master thesis. This enables indeed to reduce the mean square error to the disagreement in the system. Moreover, we will see that the lack of last message has a beneficial effect on the disagreement of the approximations in the system. Those two elements put together make the strength of this naive algorithm.

Remember the definition of the group error:

$$\mathbb{E}[(\bar{x} - \bar{r})^2] = (\bar{x} - \bar{r})^2.$$

Also, we can rewrite $\bar{x}$ as $\bar{r} + \frac{1}{n} \sum_{k=1}^{k_t} \Delta_{l_k}$ where k_t corresponds to the number of replaced agents until time step t . Δ_{l_k} therefore refers to the Δ -value associated to the k -th replaced agent when it leaves the system. As already mentioned and showed in chapter 2, those Δ -values are the basis of the group error. The average of the approximations is indeed equal to the sum the current mean of the values in the system and the perturbations due to

¹Keep in mind that we consider discrete time processes. Therefore *longer* refers here to the number of time steps. Indeed, the time steps are nothing else than the sampling of the discrete events that occur during a continuous process. Therefore, if the number of events per time unit increases with the number of agents in the continuous process, the asymptotic error, in this case, is not reached in more time for sure in the continuous process but requires more events, reflected by the discrete time process.

the lack of last message undergone so far. As we consider a (constant) normal distribution with mean 0, we can assume that the Δ_l follows a certain distribution whose mean is also equal to zero. Indeed, mostly for large systems (whose composition fits well the expectation of the normal distribution by the law of large number), the approximations of the agents are as likely to be positively or negatively influenced. Therefore, the successive replacements lead to leaving Δ to compensate previous ones or are meant to be compensated in a near future. Finally, from the sum weighted by the number of agents in the system, it requires a lot unfavorable leaving Δ - values to have a significant influence on the group error. We can then conclude that $(\bar{x} - \bar{r})^2 = \left(\frac{1}{n} \sum_{k=1}^{k_t} \Delta_{l_k}\right)^2 \approx 0, n \gg 1, \forall t$.

For small systems, since the leaving Δ have an immediate importance, even if they may get balanced over time, the group error is significant.

We can deal now with the fact that the last message strategy leads to smaller errors than the uncorrected gossip strategy in really small systems. Quickly, the uncorrected gossip strategy becomes more advantageous. In order to illustrate this group error, let us consider the case where the system closes once the MSE is stabilized (has reached its asymptotic value). Using theorem 1, we can tell that, if the system closes after a given time step, the agents will reach a consensus at a certain value c as $t \rightarrow \infty$. It comes that $c = \bar{x}_{t_c}$ where t_c corresponds to the time step at which the systems close. Therefore, the MSE is reduced to the group error when the consensus is reached. This result is represented on figure 3.7.

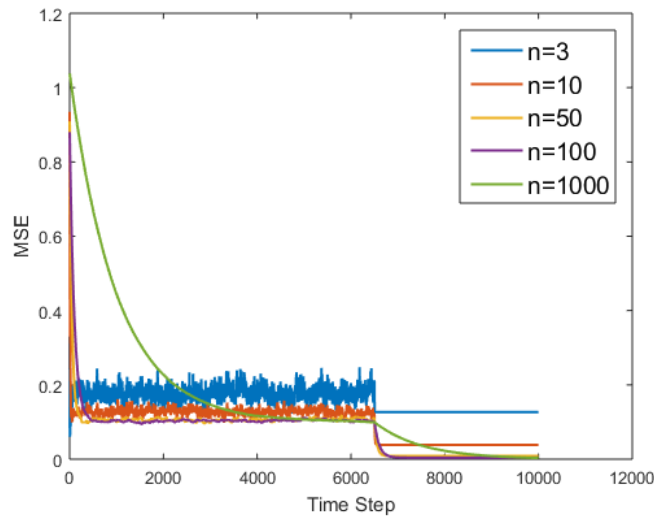


Figure 3.7: The systems get closed after 6500 time steps (uncorrected gossip strategy).

We will see in the next subsection how this group error depends on the gossip rate. Meanwhile, we can already give the asymptotic group error for a probability of gossip of .9 in table 3.1. Note that, while these values are relatively accurate for large systems, they are not for small systems due to the comments made previously about the smoothness of the curves. Anyway, it remains clear that the group error represents a weak term of error in systems larger than 50 agents (lower than $1e - 2$).

To conclude this part devoted to the dependency on the number of agents in the system, let us consider a smaller probability of gossip. Let us choose for example a gossip rate equal to .6. Those results are represented for both strategies on figure 3.8. We will not explain here the characteristics linked with the gossip rate but rather explain how the

n	Group Error
3	0.10
10	0.03
50	0.009
100	0.005
1000	0.003

Table 3.1

systems deal with this scenario according to their size.

Indeed, we have seen that for large probabilities of gossip, the more agents in the system, the lower the MSE in the uncorrected gossip strategy. This is not true anymore. This is due to the decrease of the group error. Indeed, more replacements mean that the group error is constantly correcting itself. Therefore, even for such a gossip rate, the agents are able to do a few gossips decreasing importantly the disagreement. On the other hand, the last message strategy is linked to a similar behavior. Note already the emergence of a minimum for the MSE in large systems. This will be explained in the following subsection where this phenomenon is more clear.

We also notice that the asymptotic values for the different sizes of the system seem more distant from one another. We will see it in details in the section devoted to the asymptotic values.

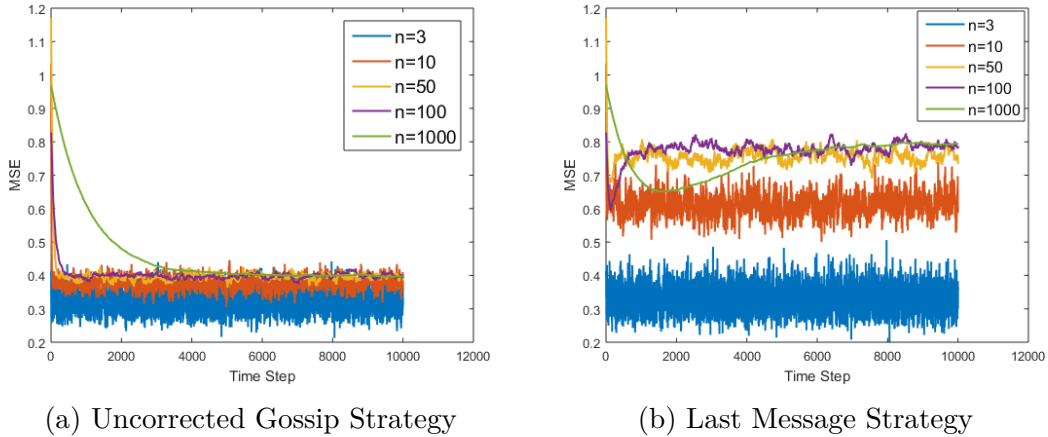


Figure 3.8: Mean evolution through time of the MSE for a probability of gossip of .6 for several sizes of systems.

You can find in Appendix A the evolution of the group error and the disagreement function of time in the uncorrected gossip strategy for both considered gossip rates.

Gossip Rate Dependency

Let us now analyze the dependence on the gossip rate. Given the results coming from the analysis of the dependence on the number of agents we will consider small (3 agents), medium sized (10 agents) and large (500 agents) systems. Those qualifiers are based on the

importance taken by the group error in the different cases. Over 500 agents, the systems have similar characteristics but require more time to be analyzed.

Before getting any further, note already that the cases where the probability of gossip is equal to 0 (only replacements are allowed) and where it is equal to 1 (closed system) lead to the same results in both strategies. Indeed, their difference stands in the presence of the announcement of the departure of the agents. If the system is closed, there is no replacement and therefore identical results come out of both strategies. However if the gossips are avoided, the agents do not change their approximation during their time in the system and broadcast thus null last messages, which is equivalent not to send any, and leads to the same conclusion. Note also that there comes therefore a null group error in both cases in the uncorrected gossip algorithm.

The mean error for those systems and several values for the gossip rate are represented on figure 3.9 to 3.11.

First, let us consider the case where the system is composed of only 3 agents. In that scenario, the error in the uncorrected gossip strategy reaches a minimum after a few time steps for large gossip rates (most clear for gossip rates equal to .9 and .95). Before the first replacement, there is no group error in the system and the disagreement decreases. Once this replacement is made, the group error starts increasing (assuming each agent has changed its approximation) until it reaches a maximum, its asymptotic value. In that first system, the last message strategy provides way better results for large probabilities of gossip. This behaviour was already represented in section 3.1 devoted to a first approach. The justification of previous subsection about the agreement found between two replacements can again be used here. However, when the gossip rate decreases, the uncorrected gossip strategy takes over. Indeed, for $p_{gos} = .6$, the results coming from both strategies are similar and the error for $p_{gos} = .2$ are a bit lower in the uncorrected gossip strategy. You can find the evolution of the group error and the disagreement on figure A.3 and A.4 in appendix A.

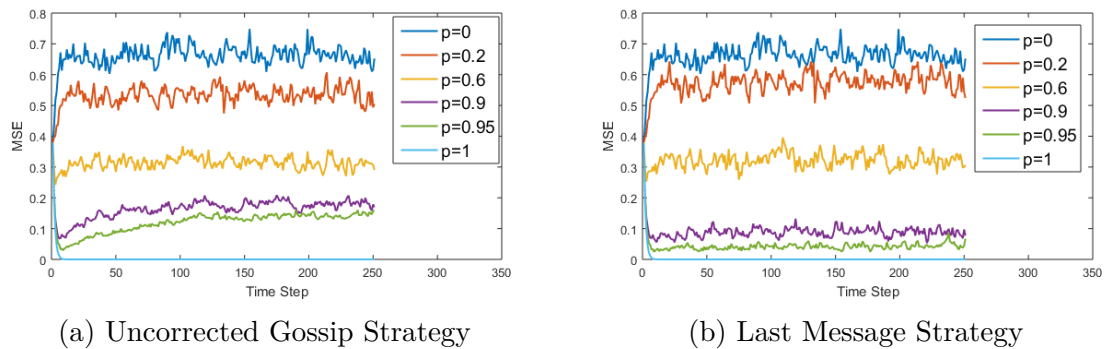


Figure 3.9: Mean evolution through time of the MSE for 3 agents in the system for several gossip rates.

If 10 agents compose the system, the last message algorithm does not seem to provide lower errors anymore, even for large gossip rates. Moreover, the results become quickly worse in that strategy and for $p_{gos} = .6$, it leads to an MSE that has a difference of around .2 with the uncorrected gossip algorithm. It represents an error 150% larger. We also observe a phenomenon that does not occur for systems containing 3 agents: the error linked with a small (non zero) gossip rate (i.e. $p_{gos} = .2$) becomes larger than the error of

a system without gossip. That phenomenon occurs in the last message strategy. In other words, agents would better not gossip at all rather than gossip and send last messages (for small p_{gos}). Therefore they should keep in such case their approximation as being their own value, whatever the outcome of an interaction. This outlines the bad efficiency of the last messages for small gossip rates.

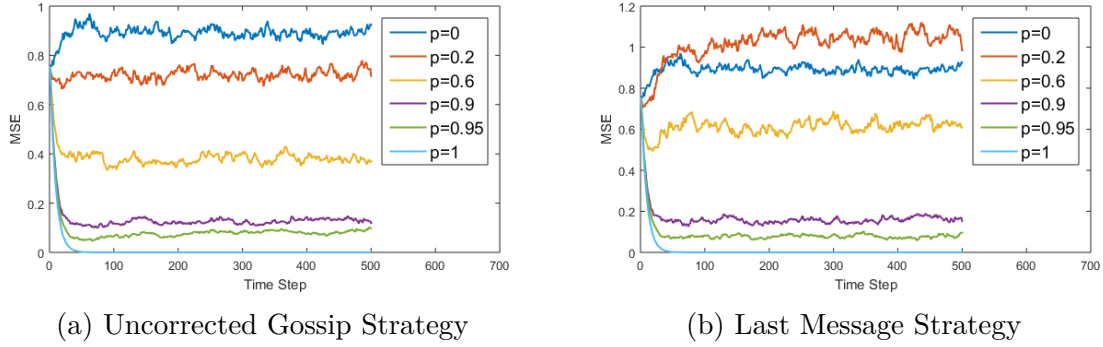


Figure 3.10: Mean evolution through time of the MSE for 10 agents in the system for several gossip rates.

When the size of the system increases again and reaches 500 agents, the minimum reached in the uncorrected gossip strategy has disappeared. Now, this minimum takes place in the last message strategy. This phenomenon already appeared in the system composed with 10 agents but is more obvious for larger systems. The explanation of this minimum is that, during the first steps, the two strategies are similar. Indeed, from the large size of the systems and the small gossip rates, the first agents that leave the system broadcast null last messages. The first few leaving agents are not likely to have gossiped so far and therefore change their approximation. However, once a significant part of the agents in the system has gossiped or received a non-zero last message, they will, in turn, broadcast non-zero messages to announce their departure.

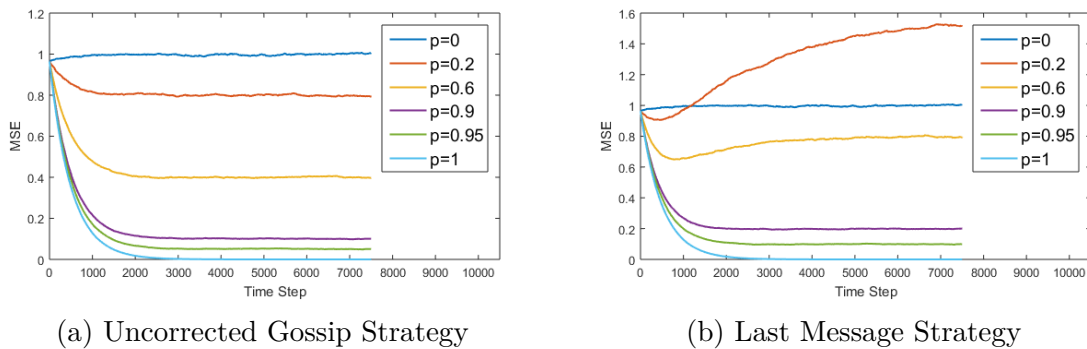


Figure 3.11: Mean evolution through time of the MSE for 500 agents in the systems for several gossip rates.

We can see it has a *snowball effect*. Initially, no message is sent by the leaving agents (due to the lack of evolution of their approximation during their time in the system) but little by little, two simultaneous phenomena happen that quickly lead agents from mostly not sending any message to most agents broadcasting a non-zero Δ - value as they leave. On the one hand two approximations are likely to be influenced during each gossip step. On

the other hand, once the agents begin to leave the system with a non-zero Δ – *value*, they become responsible for the evolution of the approximation of an agent in the system.

Also, for the uncorrected gossip strategy, the gossip rates of 0 and 1 represent respectively an upper and lower bound of the error. Indeed, it seems obvious that it is not possible to get a lower error than the one linked with a closed system (same comment for the last message strategy) since each interaction decreases the MSE² (no group error and decrease of the disagreement). For the upper bound, we can say that the less the agents gossip, the larger the disagreement in the system since the replacements is usually a source of disagreement that is then weakly decreased by gossips. Also, since large systems are linked with insignificant group errors, the MSE has a maximum for $p_{gos} = 0$ (in which case the group error is null). For smaller systems, the same characteristics come out. The explanation is however slightly different. Indeed, on the one hand, the group error seems to be increasing with the gossip rate. Therefore small probabilities of gossip imply low group error. On the other hand, when a gossip occurs, it enables to strongly decrease the disagreement.

3.2.2 Analysis of the Asymptotic Errors

Now that we have observed the behavior throughout time of a mean process for several sizes of systems and gossip rates, it is now time to consider the asymptotic characteristics of the errors according to those parameters. Some of the results given in that subsection have already been partly introduced before. This part is important because we have seen that the system error reaches quite quickly this value. Hence, if a process lasts for a long period, without any evolution of the parameters, the error in the system is expected to be most of the time around a certain corresponding value.

In this subsection, we will again consider the mean value of the asymptotic MSE as well as its different sources of error (for the uncorrected gossip strategy). Those terms are computed from the mean of the last 200 time steps for a large number of processes (depending on the size of the systems in order to reach accurate results). Also, we will consider the mean of the deviation over those 200 time steps. In other words, we compute $s_t = \mathbf{E}[(\mathbf{MSE}_t - \overline{MSE}_t)^2], \forall t \in]t_{max} - 200, t_{max}]$ and then mean it over the time steps $\frac{1}{200} \sum_{t=t_{max}-200}^{t_{max}} s_t$. In those expressions, $\mathbf{MSE}_t$ corresponds to the set of MSE in the different simulations at time step t , and $\overline{MSE}_t$ is the mean MSE at time t .

Asymptotic Error Function of the Gossip Rate

Let us begin this section devoted to the asymptotic values with the analysis of the evolution of the asymptotic error function of the gossip rate. We will outline the long term behavior of the MSE in both algorithms as well as the asymptotic characteristics of the group error and disagreement in the uncorrected gossip strategy. We have already seen that the MSE linked with the latter increases from 0 for a closed system to an upper bound $MSE_{max}(n)$ (in the uncorrected gossip strategy) when gossips are not allowed. We will indeed have a

²Remembering the comments given earlier, the MSE decreases if the gossiping agents have different values.

first hint in this section that this upper bound depends on the number of agents. These results are represented on figure 3.12 (a).

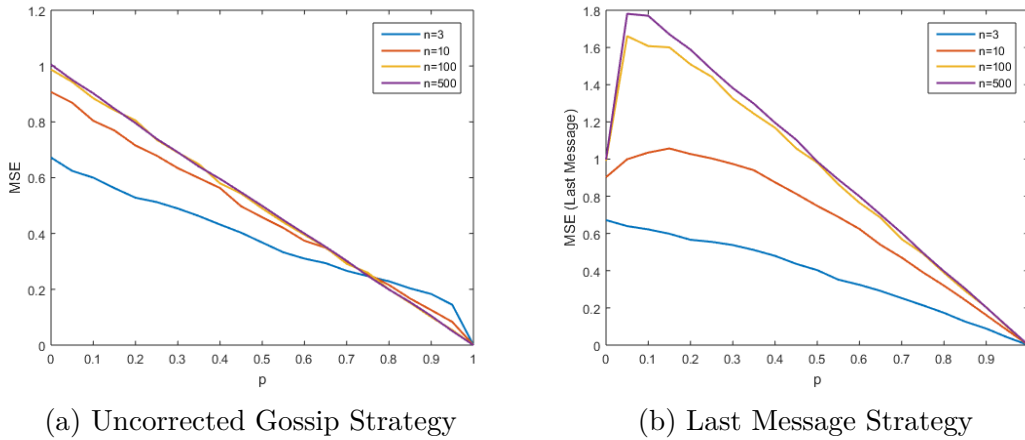


Figure 3.12: Asymptotic MSE function of the gossip rate

First, when agents keep on joining and leaving the system, the asymptotic disagreement (that is equal to the MSE given that there is no group error for such a gossip rate) is not equal to 1 (variance of the system) but tends toward $\frac{n-1}{n}$. This is explained in [So08]. Indeed, the MSE is equal to

$$\mathbb{E}[(\mathbf{x} - \bar{r})^2] = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{r})^2.$$

However, the 'correct', unbiased, sample variance should be calculated as

$$\hat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{r})^2.$$

It comes that

$$\mathbb{E}[MSE] = \frac{n-1}{n} \sigma^2 \neq \sigma^2.$$

This result explains that, for large systems, the MSE is close to the variance of the distribution ($\lim_{n \rightarrow \infty} \frac{n-1}{n} = 1 \Rightarrow \mathbb{E}[MSE] \rightarrow \sigma^2$). For more details, please refer to [So08].

We can see that the MSE decreases nearly linearly from $MSE_{max}(n)$ (for $p_{gos} = 0$) to 0 (for $p_{gos} = 1$) for large systems. Actually, this statement seems to be true for the disagreement term, whatever the size of the system. Indeed, we can also observe that the asymptotic group error increases with the gossip rate. However, while this term is insignificant for large systems, given its small size compared with the disagreement term, it has to be taken into account for small ones. Therefore, the MSE can be approximated by the disagreement in the former case but not in the latter one. Nonetheless, the asymptotic group error also increases linearly but undergoes a discontinuity in $p_{gos} = 1$. This explains the linear decrease of the MSE that tends toward the corresponding group error as the gossip rate tends toward 1 before reaching 0 for $p_{gos} = 1$. Note that the curves are drawing for p_{gos}

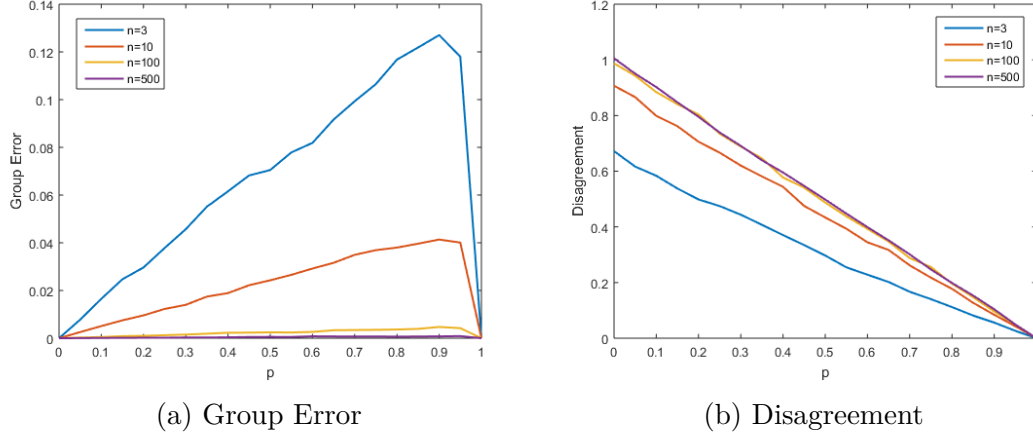


Figure 3.13: Source of errors from the uncorrected gossip strategy function of the gossip rate.

spaced by .05 from one another. This phenomenon is represented on figure 3.13. This phenomenon seems still confirmed for closer sampling as the gossip rate approaches 1.

Let us now deal with the last message strategy. We have already seen that high probabilities of gossip lead to a zero MSE but low ones lead to really bad results. Those observations are indeed represented on figure 3.12 (b). The asymptotic behavior for a null or unitary gossip rate is obviously the same as in the uncorrected gossip strategy as already mentioned. However, compared with the uncorrected gossip strategy, the decrease is not linear anymore and the MSE even reaches a maximum for a gossip rate in the interval $[0, .2]$ depending on the size of the system. This maximum tends to be coming from gossip rates closer to zero as the system grows.

Also, when the system grows, the maximum of the error gets close to 1.8 for a gossip rate of .05. From this point, the error seems to decrease almost linearly according to $MSE(p_{gos}) = 2 \cdot (1 - p_{gos})$. For small systems, this observation does not fit the results. It appears that the error decreases faster as the gossip rate increases compared to the uncorrected gossip strategy since the MSE function of the gossip rate seems to decrease linearly according to $MSE(p_{gos}) = 1 - p_{gos}$ in the uncorrected gossip strategy (large systems). It may also explain why the uncorrected gossip strategy leads to smaller errors in large systems, even when the gossip rate is important.

We observe that the value of this maximum strongly depends on the number of agents in the system. While for small probabilities of gossip the curves are clearly distinct from one another, for large gossip rates, the curves are getting closer to one another which means that the systems have similar asymptotic characteristics when the system becomes large. Let us finally consider the deviation between the results represented on figure 3.14. The MSE of the uncorrected gossip strategy is obviously linked with the larger deviations in small systems where each event is strongly influencing the error. This large deviation between the processes is due to a lot of simulated processes required to obtain smooth enough results for small systems. Also, the deviation between the processes is a decreasing function of the gossip rate. When the gossip rate is small, the MSE of a process is frequently evolving due to the replacements. When it increases, the systems are able to decrease the error thanks to gossips which lead the errors closer to one another. Consider

indeed a gossip rate close to 1, then most of the simulated processes are linked with a disagreement close to zero most of the time. They are however causing a group error that may be significant and may therefore be importantly different from one simulation to another. This explains why the deviation is equal to 0 for $p_{gos} = 1$ and still relatively far from it for $p_{gos} = .95$ in small systems.

In the last message strategy, the deviation between the errors in the different processes is much larger than in the uncorrected gossip strategy (except for $p_{gos} = 0$ and $p_{gos} = 1$ where the two strategies are identical). This is explained on those graphs by the regular decrease in the uncorrected gossip strategy while the results for the last message algorithm are initially increasing. This error is even more obvious in large systems where the deviation between the processes is insignificant in the uncorrected gossip strategy. This comes from the fact that the messages sent in the system are linked to variable errors that remain for a possibly long time in the system.

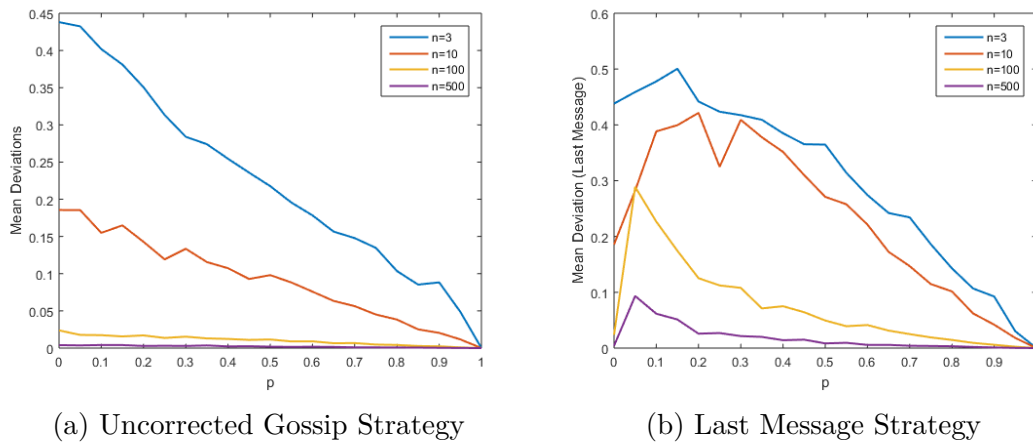
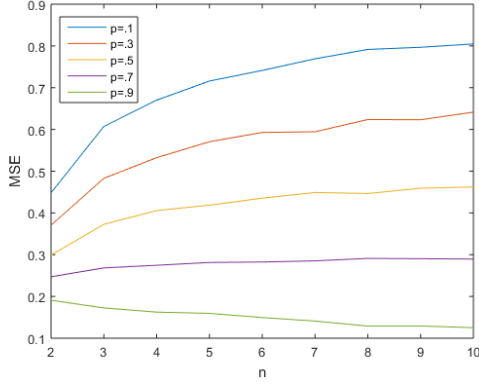


Figure 3.14: Asymptotic mean deviation function of the gossip rate for both strategies.

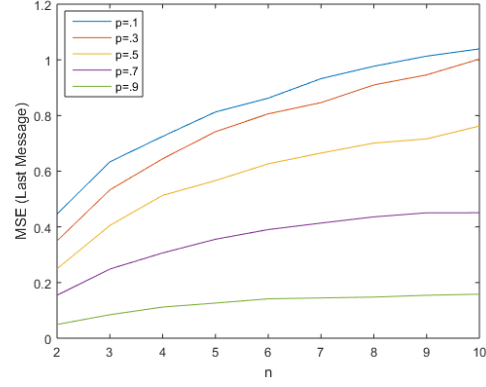
Error Function of the System Size

Let us conclude this section with the analysis of the asymptotic values as a function of the number of agents in the system. This asymptotic MSE in both strategies is represented on figure 3.16. As seen in the subsection devoted to the analysis of the process function of the number of agents in the system, we will limit the analysis to systems composed of 10 to 100 agents since larger systems present close asymptotic properties to systems composed of 100 agents. We will also zoom on systems composed of 2 to 10 agents (figure 3.15). This seems indeed to be the area where the MSE is most influenced. Note that systems composed of 1 agent do not make any error since there is no disagreement nor group error in any of the two strategies ($\bar{x} = x_1 = r_1 = \bar{r}$).

Actually, we can see that the evolution of the error for increasing numbers of agents in small systems is in the continuity of those in larger ones. Indeed, the curves present the same behavior except that the evolution is more important. For example, for the uncorrected gossip strategy, the MSE decreases from .19 for 2 agents to .125 for 10 agents to finally .105 for 100 agents ($p_{gos} = .9$). For $p_{gos} = .1$ it increases from .45 for 2 agents to .805 for 10 agents to .885 for 100 agents. The analyses of the graphs for smaller and larger systems are therefore similar and we will therefore not comment them separately.

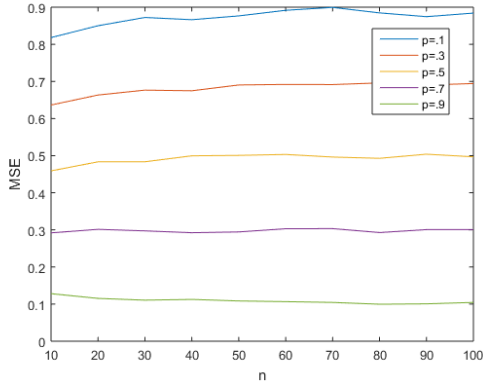


(a) Uncorrected Gossip Strategy

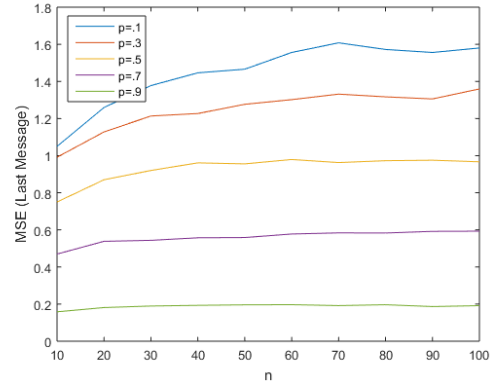


(b) Last Message Strategy

Figure 3.15: Asymptotic MSE function of the system size for both strategies in small systems.



(a) Uncorrected Gossip Strategy



(b) Last Message Strategy

Figure 3.16: Asymptotic MSE function of the system size for both strategies.

In the results corresponding to the uncorrected gossip strategy, the curves of the asymptotic MSE in function of the number of agents decrease only for large gossip rates. This is due to the fact that the decrease of the group error is significant compared with the evolution of the disagreement term in that case (see figure 3.18). Indeed, as we have seen in the part devoted to the dependence of the process on the gossip rate, the group error is lower for small probabilities of gossip. Indeed, for similar gossip rates, the agents are more slowly replaced in large systems.

Also, as n grows, the asymptotic MSE tends toward $1 - p_{gos}$. This seems indeed to be confirmed by figure A.5. This figure represents the evolution of the variance during a process of a system composed of 200 agents. Those graphs are drawn thanks to the steady state equation given in [HM17]³. Remember indeed that, in large systems, the group error is insignificant and the MSE can be reduced to the disagreement.

Let us now consider the results coming from the last message strategy. We can see on those two figures that the asymptotic MSE increases with the size of the system, whatever the

$${}^3\mathbb{E} \left[\frac{\bar{x}^2}{x^2} \right] = \begin{bmatrix} 1 - 2 \frac{1-p_{gos}}{n} & \frac{1-p_{gos}}{n^2} \\ \frac{p_{gos}}{n} & 1 - \frac{1}{n} \end{bmatrix} \mathbb{E} \left[\frac{\bar{x}^2}{x^2} \right] + \sigma^2 \left[\frac{1-p_{gos}}{n^2} \right]$$

probability of gossip. We observe however that the MSE is more sensitive to the number of agents in the system for low probabilities of gossip. Indeed, for $p_{gos} = .1$, $p_{gos} = .9$ the MSE evolves from 1.04, 0.16 in a system composed of 10 agents to 1.58, 0.19 for a system composed of 100 agents, which represents a difference of 0.54, 0.03. The increase of the asymptotic MSE as the system grows comes from the, already explained, increase of the disagreement.

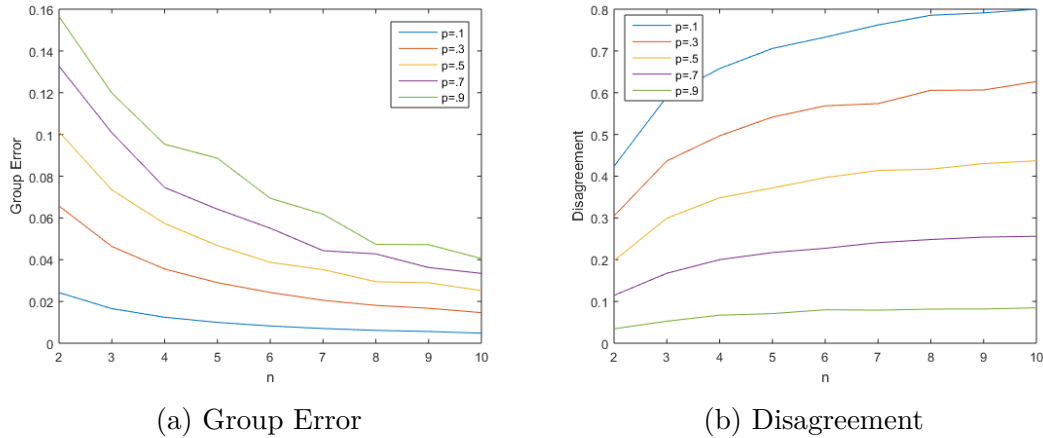


Figure 3.17: Asymptotic MSE function of the system size for both strategies in small systems.

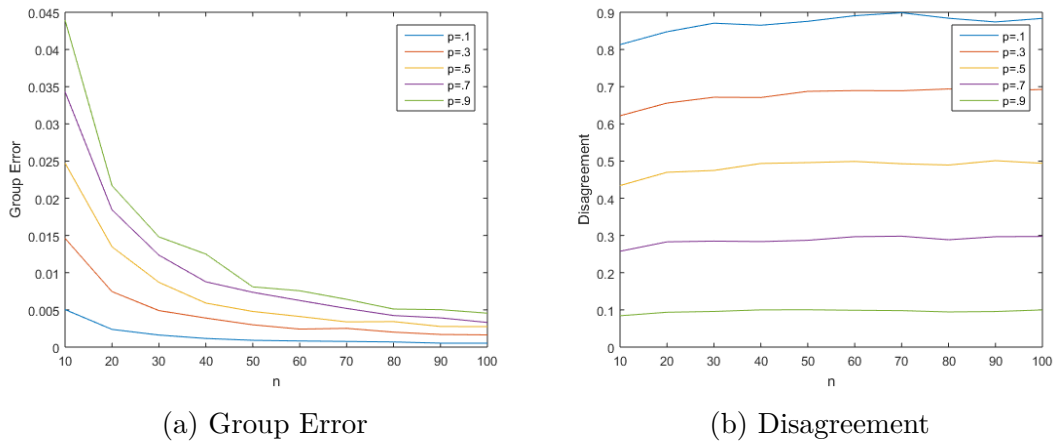


Figure 3.18: Asymptotic MSE function of the system size for both strategies.

Let us conclude that first analysis of the asymptotic error function of the size of the system with the mean deviation during the last 200 time steps between the mean MSE at each time and the MSE of each process for the corresponding times. This result is represented on figure 3.19. Once again, the uncorrected gossip strategy is ahead since the corresponding deviation is lower than in the last message strategy. Note that the graduation of the graphs may be misleading in the analysis of the two algorithms.

Note also that this deviation associated to the results of the uncorrected gossip strategy is initially quickly decreasing as the size of the system increases so that the space between the curves corresponding to the different gossip rates is quickly reduced.

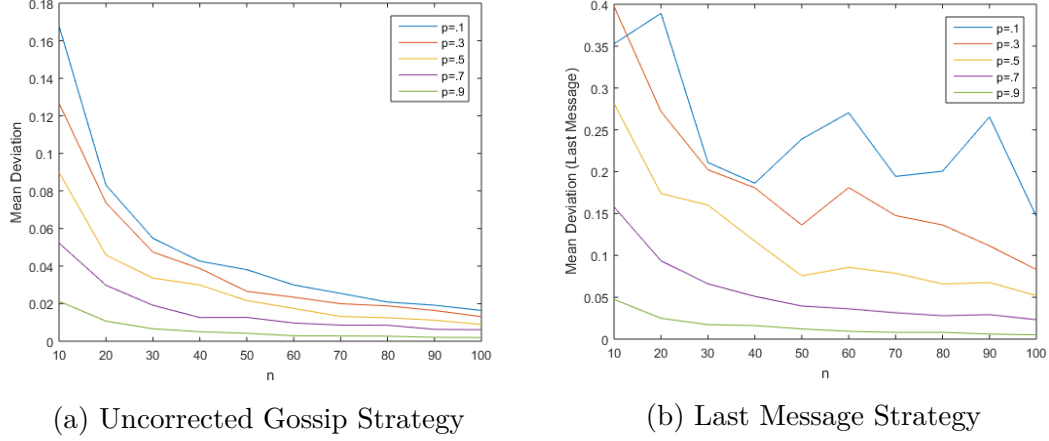


Figure 3.19: Asymptotic MSE function of the system size for both strategies.

3.3 Time-Depending Distribution of the Incoming Values

Let us now consider the scenario where the mean of the distribution of the incoming values $\mu(t)$ evolves through time. We will consider that it increases linearly between two bounds that are 0 and μ_{max} over a certain number t_{max} of time steps. $\frac{\mu_{max}}{t_{max}}$ therefore designates the rate at which $\mu(t)$ increases. We will analyze the situations where this ratio is small or large and its influence on the error according to the gossip rate and the size of the system.

3.3.1 Evolution of the Mean of the Values in the System

Before analyzing the error, let us first observe the effect of that perturbation on the mean of the values in the system. This evolution for $\mu_{max} = 1$ is represented on figure 3.20 considering a constant gossip rate (considering $t_{max} = 5000$) or a fixed size for the system (considering $t_{max} = 500$).

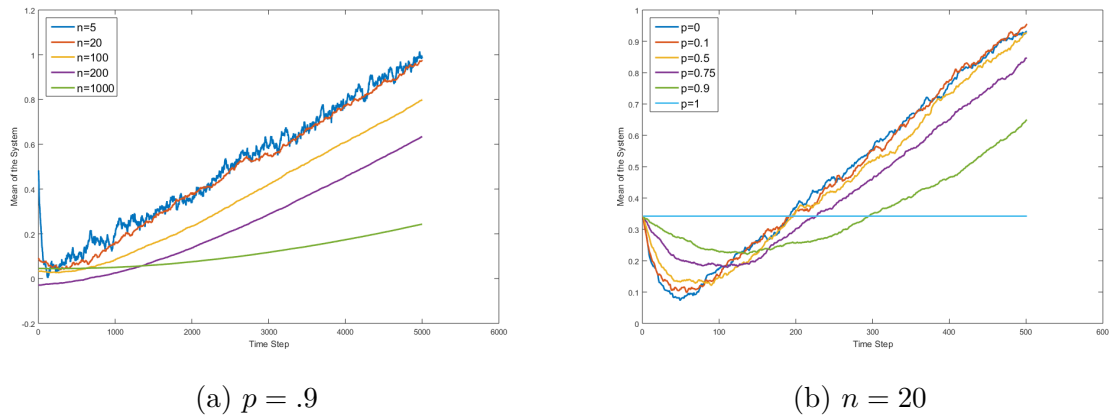


Figure 3.20: Mean of the system where the distribution of the values coming into it is linearly increasing through time with $\mu_{max} = 1$ over 5000 time steps.

The conclusions are somehow trivial but are interesting to visualize in order to interpret

the results presented in this section. If the gossip rate is fixed, the evolution of the mean of the system grows more quickly for small systems. Given the long period of time during which the process takes place, the mean of small systems tends to follow the mean of the distribution. Indeed, in a small system, as the agents are more frequently replaced, the system is somehow constantly renewed in the sense that only a few time steps are required to replace all the agents in the system. From the slow evolution of $\mu(t)$, its value is not likely to have significantly grown as the system gets renewed, which explains the evolution of the mean in that scenario. The slow evolution of the mean of large systems is therefore coming from the fact that an incoming value has less weight and therefore lot more replacements are required to have the same effect as in small systems.

From that first graph, we can already have a first intuition about the upcoming results. For such a slow evolution of the mean of the system, small systems will not feel the evolution of $\mu(t)$. Indeed, it is likely that two replacements are only a few time steps apart from each other for $p_{gos} = .9$. The increase of $\mu(t)$ at a new replacement is therefore not significant. The agents in such a small system therefore interact as if the mean is constant during their time in the system. The messages sent or errors made should therefore not explode.

If the system grows, the mean of the system is increasing more slowly. Therefore, in that situation, agents that enter the system in late times are likely to have a value larger than the mean of the system. They may therefore leave the system with large negative Δ - values. However, most agents leave the system with a positive Δ - value. It might therefore once again lead to small group errors associated to the uncorrected gossip strategy and therefore to good results associated with them.

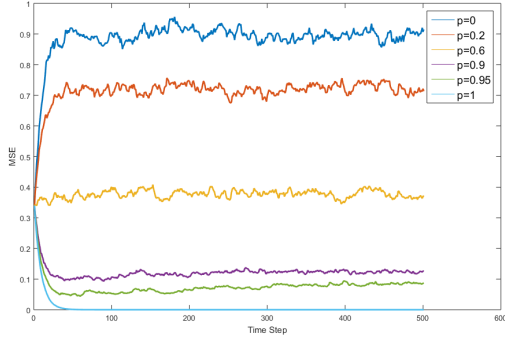
Similar comments can be made concerning the gossip rates where small gossip rates imply a lot of replacements and can therefore be assimilated to small systems where the $\mu(t)$ does not increase much between two replacements. Large gossip rates behave like large systems where $\mu(t)$ may be importantly increasing between two replacements and where the mean of the system does not follow the mean of the distribution and therefore the mean of the system depends on values that entered the system a long time ago and that are likely to be smaller than the current mean of the distribution.

Note also the initial decrease of the mean that is mainly observable in small systems. On figure 3.20 b), the mean is initially different from zero. The first replacements are normally distributed with a mean close to 0, which is below the initial mean of the system. This explains the initial decrease in small systems (with an initial mean larger than zero) which is less important in larger systems since the decrease has been seen to be slower. Therefore, from the time $\mu(t)$ equals the mean of the system, the mean of the system starts increasing again. Note also that the mean of the system is obviously constant in a close system.

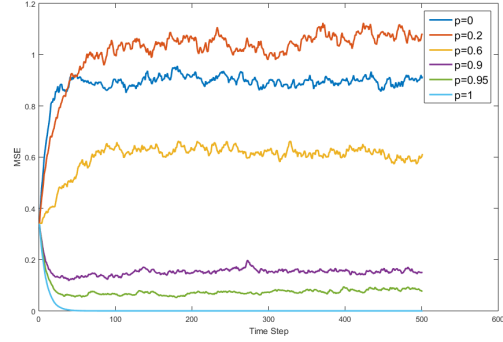
3.3.2 Uncorrected Message and Last Message Strategies

Now that we have observed the evolution of the mean of the values in the system, we can analyze the results of the errors in both strategies. The evolution of the error for both the uncorrected gossip and last message strategy are represented on figure 3.21 for systems containing 10 agents and figure 3.22 for systems composed of 100 agents. Those will be respectively referred as small and large systems. These graphs represent the evolution of the error for several gossip rates considering a constant, slowly increasing and a quickly

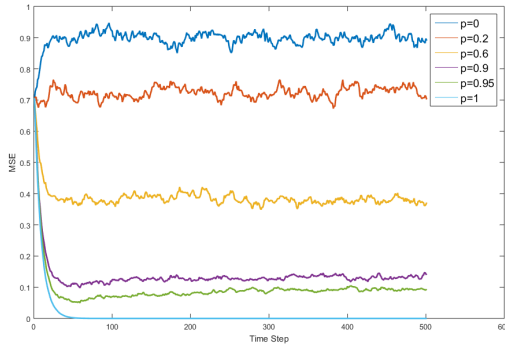
increasing mean of the distribution over the same lap of time.



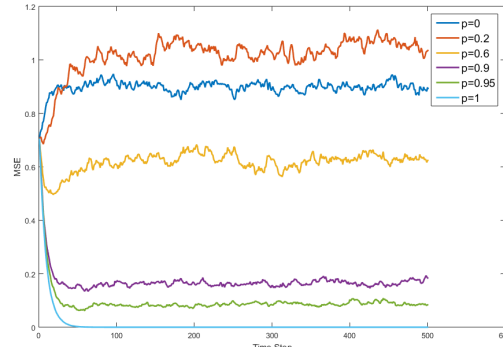
(a) Uncorrected Gossip Strategy, $\mu = 0$



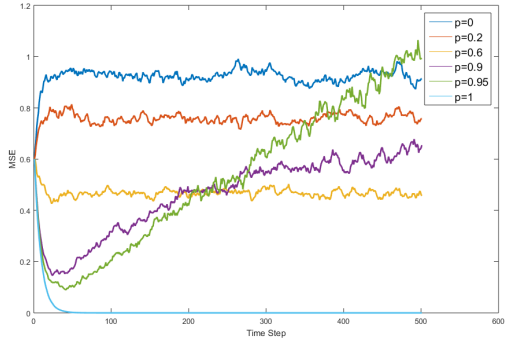
(b) Last Message Strategy, $\mu = 0$



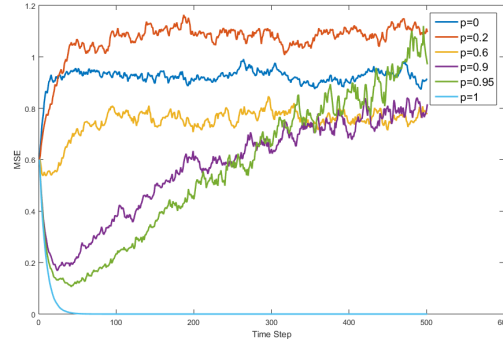
(c) Uncorrected Gossip Strategy, $\mu_{max} = 1$



(d) Last Message Strategy, $\mu_{max} = 1$



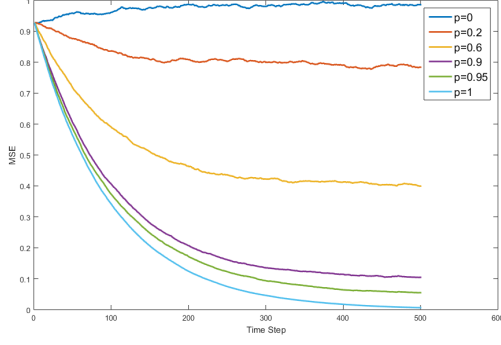
(e) Uncorrected Gossip Strategy, $\mu_{max} = 10$



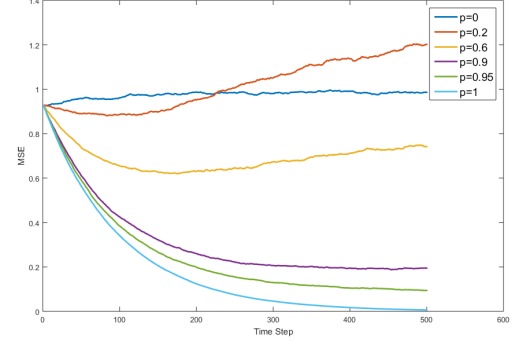
(f) Last Message Strategy, $\mu_{max} = 10$

Figure 3.21: Evolution of the MSE through time for 10 agent systems considering a different evolution of the mean of the distribution in both strategies.

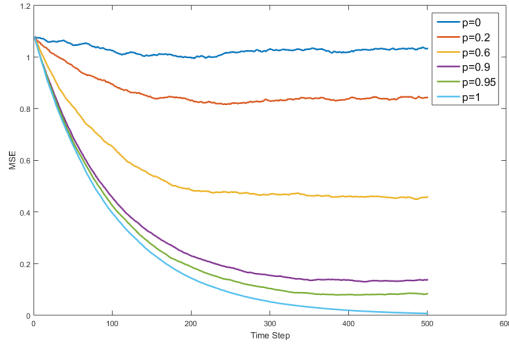
Let us first observe the behavior of the error in small system. A first observation is that when the agents are constantly replaced (without gossiping at all), the error is similar to the constant distribution problem in the case where the distribution evolves, even for relatively large μ_{max} . This comes from the explanation stated above that the mean in small systems closely follows the mean of the distribution. Also, as the mean of the distribution is close to the mean of the system, the disagreement due to the replacement is not fundamentally different from the scenario of constant distribution. However, when the system grows, values that may be way larger than the current mean of the system join it, which results in a large disagreement. This error remains in the system until those agents get replaced by an agent having a value closer to the mean.



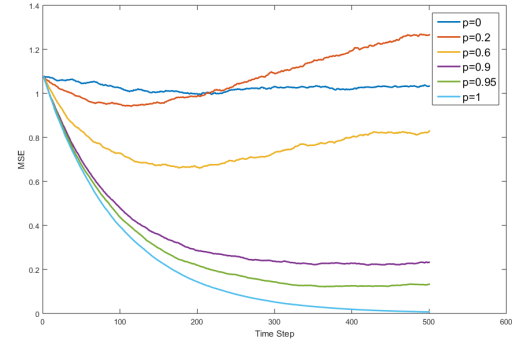
(a) Uncorrected Gossip Strategy, $\mu = 0$



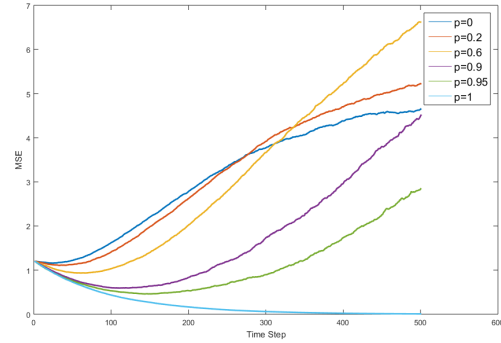
(b) Last Message Strategy, $\mu = 0$



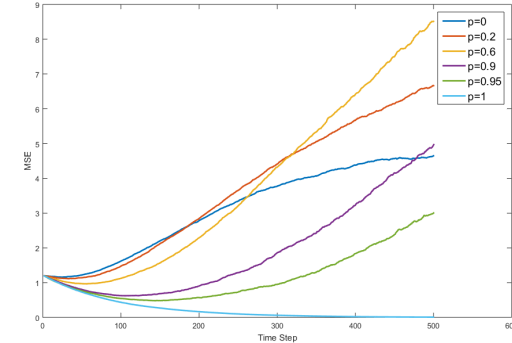
(c) Uncorrected Gossip Strategy, $\mu_{max} = 1$



(d) Last Message Strategy, $\mu_{max} = 1$



(e) Uncorrected Gossip Strategy, $\mu_{max} = 10$



(f) Last Message Strategy, $\mu_{max} = 10$

Figure 3.22: Evolution of the MSE through time for 100 agent systems considering different evolutions of the mean of the distribution in both strategies.

As expected, we see that the MSE in the uncorrected gossip strategy leads to similar results in the cases where the mean is constant or weakly increases. Hence the agents deal with incoming values as if they were normally distributed with a constant mean over a certain time span. When μ_{max} strongly increases and reaches 10, the error linked with large gossip rates is the most impacted by the perturbation. This results from the likely large increase of $\mu(t)$ at new replacements linking the incoming values to a possibly large disagreement. Also, the evolution of the distribution may lead the system to have a large range of values. The large gossip rate implies that agents reach a consensus. The agents in the system may therefore strongly change their approximation during their time in the system, linking them to important Δ_i as they leave the system. The group error is therefore rarely evolving, and when it does, it is corrected with possibly large Δ_i that are

therefore less likely to keep the group error close to zero. This case is therefore different from the constant distribution where the group error evolved little by little thanks to small unannounced Δ – *values*. Note that in that case, the mean group error can reach .4 at the end of the process when the $p_{gos} = 95$.

The results corresponding to the last message strategy in small systems where $\mu(t)$ is weakly increasing through time is also very similar to the case where the mean of the distribution is constant. The reasons are similar to those given for the uncorrected gossip strategy. Once again, mainly the errors linked with large gossip rates are different when $\mu(t)$ increases up to 10. Of course, there is no question of group error here. The mean of the system is mostly below the mean of the distribution. Therefore, the agents coming into the system are likely to hold a value larger than the current mean of the system. A large disagreement comes therefore from the incoming values. Also, given its small size, the mean of the system is consistently influenced by replacements. Therefore, the agents may change importantly their approximation resulting in large messages, also source of disagreement.

When the system grows and contains 100 agents, if the errors are again not significantly increasing when $\mu(t)$ increases up to 1 over 500 time steps, the results are way different when the mean is strongly evolving through time. Nevertheless, when $\mu_{max} = 1$, the error is more felt in large system than in small ones, since, when $t \rightarrow t_{max}$ and therefore $\mu(t) \rightarrow \mu_{max}$, the incoming values are further from the mean of the system. This leads to a small and yet present disagreement. This phenomenon is intensified for larger values of μ_{max} for which joining values are obviously likely to be larger in magnitude. This does of course not apply to closed systems, which are not affected by replacements and therefore not by the time-dependent distributions.

In both strategies, and it was also true for smaller systems, while large probabilities of gossip originally leads to lower errors, this does not last on the long term. Initially, the values in the system are randomly distributed. The first gossips decrease this *disorder* and enable to decrease the disagreement and therefore the MSE. However, little by little, the values joining the system are strongly affecting that disagreement and the gossip steps are not sufficient to compensate this error coming from the replacements. On the other hand, for small gossip rates, the disagreement quickly starts increasing due to the increase of the mean of the distribution and the lack of many gossips. But given the high number of replacements, the mean of the system is closer to the mean of the distribution of the incoming values. The system is therefore less disturbed by replacements in later times, when they are felt the most. Indeed, for both strategies, the results are close to each other for corresponding gossip rates, whatever μ_{max} . Indeed, initially, the mean of the distribution is close to the mean of the system. Mostly for late times, the evolution of $\mu(t)$ is most damaging. Obviously, this comment depends on the rate at which $\mu(t)$ increases. The slower it increases, the longer the system is close to a system where the incoming values are given from a constant distribution.

We can finally make a last comment in this section to outline the link between the rate at which the mean of the distribution increases and the size of the system. We have seen in this section that a similar increase of μ over a similar time span has not the same effect on different sizes of systems. Indeed, we see that, on the graph corresponding to $\mu_{max} = 10$ (which is the most obvious case), the errors remain below 1.2, whatever the strategy for the considered gossip rates when $n = 10$. However, the MSE may rise over 8.5 when $n = 100$.

Indeed, when the system grows, it requires more time to be highly renewed and therefore is more impacted by a similar evolution. Therefore, to observe similar characteristics in small and large systems, the rate of increase of $\mu(t)$ should take into account the number of agents in the system.

We will not analyze the asymptotic values in this section. First of all, it depends on too many parameters. With the gossip rate and the size of the system, the rate of the evolution of $\mu(t)$ and the time during which it evolves have an influence on those errors. Moreover, the errors do not stabilize around a certain value in that case since it keeps on increasing during the last steps of the processes.

3.4 Conclusion

This chapter introduced the numerical results of both algorithms. While in theory the last message algorithm proposed promising properties compared with the uncorrected gossip strategy, the numerical results have been shown to be totally different. Two reasons explain that observation. On the one hand, the last message is beneficial against the group error but this advantage is largely offset by the unwanted effect it has on the disagreement. When a replacement occurs, two approximations in the systems are changed: one directly linked with the replaced agent and one due to the incorporation of the message into an agent's approximation. This disagreement can be handled in small systems given that the agents are quickly able to find a consensus and therefore largely decrease the MSE in a few gossip steps. In large systems, the agents are less frequently included in gossips and therefore a message received may lead to a disagreement remaining in the system for a long period of time. On the other hand, the group error that was expected to be different from zero is in fact not that far from it. When the system is composed of only 100 agents, the group errors drop already in an order of magnitude of $10e - 3$ that is already insignificant compared with the disagreement term.

Moreover, besides the smaller MSE coming from the uncorrected gossip strategy in most situations, this strategy is associated with smaller deviation between the asymptotic MSE of the different simulated processes. Therefore, a system is more likely to lead to results closer to the corresponding expected errors in the uncorrected gossip strategy than in the last message algorithm.

It remained therefore to be seen if the last message algorithm cannot take advantage of its good theoretical expectations when the mean of the distribution of the incoming values evolves with time. Once again, we had to lower our sights. When the mean of the distribution does not increase at a high rate (the notion of rate should also take into account the size of the system), the two methods barely realize this modification. However, when the distribution strongly increases through time, so do the errors. Still, the lack of announcement of the departure of an agent is not that problematic compared with the disagreement introduced in the system by the messages even if the group error may become significant in certain cases.

This chapter has shown the uncorrected gossip strategy to be ahead of the last message strategy in many situations. However, this latter algorithm still has the advantage to be associated to a zero group error that may be useful in two cases. First, if the system

closes after a certain time during which replacements occurred, it will lead to a null MSE, whatever the gossip rate during the first part of the process or the size of the system. This advantage is nonetheless mitigated for large systems for which the group error is very small in the uncorrected gossip strategy. Secondly, in systems composed of only a few agents that gossip enough, the group error is in that case large in the uncorrected gossip strategy so that it penalizes the low disagreement in that strategy. In that case, the agents are quickly reaching an agreement and therefore result in a null error using the last message strategy.

Chapter 4

Reserve-Based Algorithm

We have seen that the sudden insertion of the last message, that modifies a second approximation in the system as a replacement occurs, is a source of error. In order to avoid this undesired phenomenon, we will imagine a strategy where the leaving agents still send a message in order to keep the average of the values inside the system. However, in the contrary with the last message algorithm, the last messages, through the use of reserves, are included little by little in the system. To do so, we will consider algorithms that use reserves (or surplus), $s_i(t)$. The agents are therefore linked to three values in that algorithm. Again, they are assigned a value of their own (r_i) and are still searching for an approximation of the system (x_i). The third value is the reserve that is somehow a source of group error, since it is not involved in the approximation of the agents. This term of error is meant to disappear as the reserves are reincluded in the system. It comes that the conservation of the average is now translated by $\bar{x}(t) + \bar{s}(t) = \bar{r}(t)$.

This chapter has basically the same structure as the previous one. We will first describe the algorithm that will be studied. We will then analyze its behaviour, once again considering a constant distribution of the incoming values and one whose mean increases through the process. In the former case, we will first observe the characteristics of the algorithm through time during the process before analyzing the asymptotic values in function of the parameters of this new algorithm. One goal of this chapter is to derive appropriate values for the coefficients used in the algorithm .

4.1 Description of the Algorithm

Let us begin by the presentation of the algorithm that we will use here. Reserves have already been considered in the literature. Indeed, [CI12] propose a surplus-based algorithm meant to lead the agents toward an average consensus. Our problem is however different from the one considered in this article. Indeed, we treat pairwise, bidirectional, gossips among the total set of agents in the system while this paper consider that the agents in the system form a non complete directed graph. [CI12] also consider that an agent may send information to more than one other agent during a gossip step. Above all, the main difference is that this article deals with closed systems. In [CI12] surpluses are coming from the gossip steps while we will consider in that chapter that they are only

coming from the replacement steps. The gossips are here used to include them back into the system.

Let us first now describe the two events in this algorithm, beginning exceptionally by the replacement step that is the source of the reserves.

Replacement: Two uniformly randomly agents $l, m \in \mathcal{S}(t)$ are selected. The leaving agent l computes its Δ -value that is $\Delta_l(t) = x_l(t) - r_l$ and sends it to the agent m as well as its current reserve $s_l(t)$. The agent m includes this message according to a coefficient k_2 , and stores the rest of it in its own reserve with the surplus of the leaving agent, such that

$$x_m(t+1) = x_m(t) + k_2 \Delta_l(t), \quad s_m(t+1) = s_m(t) + s_l(t) + (1 - k_2) \Delta_l$$

Agent l then leaves the system and is replaced by a "new" agent $e \notin \mathcal{S}(s)$, $\forall s \leq t$, that joins the system. Therefore $\mathcal{S}(t+1) \neq \mathcal{S}(t)$ but $n(t+1) = n(t) = n$. ■

Algorithm 4 Reserve-Based Algorithm: Replacement Step

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
 Select uniformly randomly two agents $l, m \in \mathcal{S}(t)$
 $\Delta_l(t) = x_l(t) - r_l$
 $x_m(t+1) = x_m(t) + k_2 \Delta_l(t)$
 $s_m(t+1) = s_m(t) + s_l(t) + (1 - k_2) \Delta_l(t)$
 New agent $e \notin \mathcal{S}(s)$, $\forall s \leq t$ ($r_e = x_e(t)$)
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

Gossip: Two agents $i, j \in \mathcal{S}(t)$ are uniformly and randomly selected among the n agents present in the system, and they update their approximation and their reserve by performing a pairwise average:

$$x_i(t+1) = x_j(t+1) = \frac{x_i(t) + x_j(t)}{2} + k_1 \frac{s_i(t) + s_j(t)}{2},$$

$$s_i(t+1) = s_j(t+1) = (1 - k_1) \frac{s_i(t) + s_j(t)}{2}.$$

■

Algorithm 5 Reserve-Based Algorithm: Gossip Step

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
 Select uniformly and randomly two agents $i, j \in \mathcal{S}(t)$
 $s_i(t) = s_j(t) = \frac{s_i(t) + s_j(t)}{2}$
 $x_i(t+1) = x_j(t+1) = \frac{x_i(t) + x_j(t)}{2} + k_1 \frac{s_i(t) + s_j(t)}{2}$
 $s_i(t+1) = s_j(t+1) = (1 - k_1) \frac{s_i(t) + s_j(t)}{2}$

This algorithm is associated to two coefficients that are supposed to be constant:

- k_1 is the *coefficient of reinsertion*. It corresponds to the proportion of the reserves that has to be taken into consideration in a gossip step.

- k_2 is the *coefficient of direct inclusion*. It corresponds to the proportion of the last message that has to be immediately included into an agent's approximation in a replacement step.

Those coefficients are taken in the interval $[0, 1]$ in order to keep the meaning of the last message. Indeed, a negative value for the coefficients means that the agents increase the magnitude of the reserves. A value larger than 1 means that the agents use more than they have (inclusion in the system of a value larger than the reserve held or message received which loses its meaning). A goal is therefore to choose properly the coefficients.

Concerning the gossip steps, we consider the case that leads the gossiping agents to share their surplus as well as their current approximation of the system. The main reason is that the sum of the square of the reserves after the interactions is lower. Indeed, if X, Y correspond to the reserves of the two agents before a gossip, the square of the sum of these reserves after the interaction is therefore equal to

$$2(1 - k_1)^2 \left(\frac{X + Y}{2} \right)^2 = (1 - k_1)^2 \frac{X^2 + Y^2 + 2XY}{2}$$

while it is equal, in the case where agent do not put their reserve entirely in common, to

$$(1 - k_1)^2 (X^2 + Y^2) \geq (1 - k_1)^2 \frac{X^2 + Y^2 + 2XY}{2}.$$

The equality being reached when the reserves are the same. In other words, it avoids as much as possible to have disproportionate magnitudes for the reserves. Imagine for example that $X = -Y$. In that case the outcome of the gossip will be, in both, to add "0" in the interaction coming from the reserve. In the chosen strategy, $X, Y \rightarrow 0$ while in the other one, $X \rightarrow (1 - k_1)X \neq 0$, $Y \rightarrow (1 - k_1)Y \neq 0$. If it changes nothing in the group error (mean of the reserves is still the same), it may be an advantage for the further gossips. Indeed, when a gossip occurs, the reserves that are taken in the approximations of the system may be a source of disagreement. The reserves are designed to smooth the inclusion of the last messages and therefore to decrease the associated disagreement. It seems therefore appropriate to split the reserves as much as possible so that the coming gossips will be less impacted by those inclusions, and thus less by the last messages. After a gossip, agents have therefore the same approximation of the mean of the system and the same surplus.

Note already that, if $k_1 = k_2 = 0$, the reserve based algorithm corresponds to the uncorrected gossip strategy (the last message is neither used in the replacement steps nor in the gossip steps). If $k_2 = 1$, this strategy is then equal to the last message strategy (message entirely included in the replacement steps, reserves are therefore null at all times).

Note finally that, if magnitude of the reserve of an agent falls under a certain limit (fixed arbitrary) after an interaction, it is then included to its own approximation. We assume indeed that, if a reserve is close enough to zero, including it in the system leads to an insignificant increase of the disagreement. It is also a way to lead to a null group error if the system closes after a certain number of time steps.

4.2 Constant Distribution of the Incoming Values

Let us begin the analysis of the results of this reserve-based strategy with the analysis of the 2 coefficients, k_1 and k_2 , present in this algorithm. We can obviously find values for both of them that improve the results of the last message strategy since $k_1 = k_2 = 0$ corresponds to the uncorrected gossip strategy, that has been seen to be associated with lower errors in most cases. The question is rather to know whether it is possible or not to find coefficient that leads to better results than those coming from the uncorrected gossip strategy.

The analysis concerning the dependence to the two parameters (gossip rate and size of the system) has already been done in the previous section and is not repeated here. Indeed, the comments are overall the same since this reserve-based strategy lies between the uncorrected gossip and last message algorithms. We will still outline the particular behavior of this strategy given the size of the system or the gossip rate without giving much details.

4.2.1 Analysis of the Process: Influence of k_1

Let us first observe the dependence on coefficient k_1 given that we already have an intuition about coefficient k_2 . Indeed, the last message is usually a source of raise of the disagreement (remember that changing two approximations during the replacement steps seems to be the main drawback of the corresponding algorithm). Storing it entirely may be assumed to be useful so far. We will therefore observe the evolution of the error through time according to k_1 given $k_2 = 0$.

Given the good efficiency of the last message strategy to deal with small systems, we will mainly focus on systems composed of at least 10 agents. The MSE for several systems is represented on figure 4.1 for a gossip rate of .9. We can observe that, as expected, it is possible to improve the last message strategy using reserves.

Note that the case $k_1 = 1$, $k_2 = 0$, is not equal to the last message strategy. The difference is that the last message is included in the system only once an agent gets in a gossip. Therefore, a group error due to the reserves, that has been seen not to be problematic in large systems, gets in the system but the disagreement, that is the real drawback of the last message, is mitigate since the message is somehow sent to two agents instead of one. Actually, once the agents holding the message in reserve gossip, it leads to the same outcome than in the last message strategy.

In small systems with high gossip rate, a non-zero coefficient k_1 might be useful. However, as the size of the system grows, the uncorrected gossip strategy becomes again the best algorithm to use. The slow insertion of the reserves in the system appears to be useful in that sense that it decreases the error coming from the last message strategy. On the one hand because it is a way to include the last message by less affecting the disagreement. On the other hand, the group error in the reserve-based algorithm functions as in the uncorrected gossip strategy: by correcting itself. When the agents gossip, the reserves may be included in the system, leading to an increase or a decrease of the group error. However, the gossip step tends to decrease the magnitude of the reserves. If the gossip

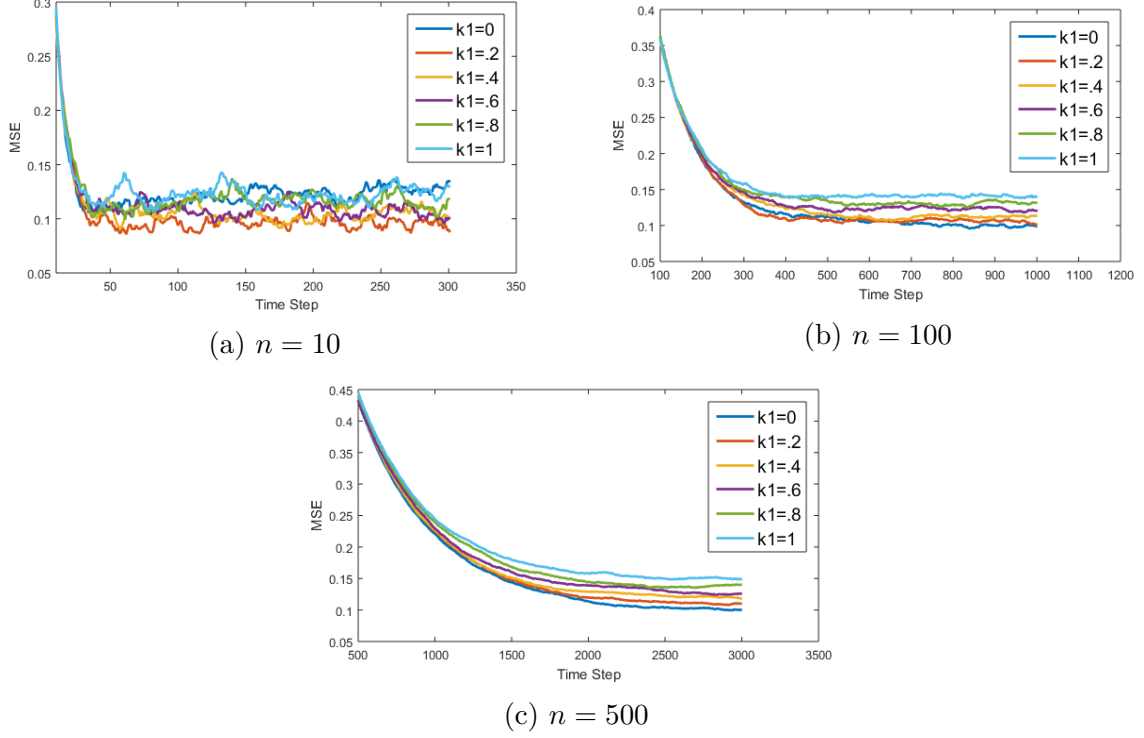


Figure 4.1: Evolution of the MSE in an open system for a gossip rate of .9 considering different sizes of system. k_2 is set to 0 and several values for k_1 are considered.

rate is important, the reserves are then quickly smoothed and the group error is then weakly influenced by a gossip step. Their magnitude in that case may also become small. The evolution of the group error is represented on figure 4.2 for system composed of 10 and 100 agents, considering $k_1 = .2$, $k_2 = 0$. This figure indeed shows that the group error is around five times smaller in this reserve-based algorithm than in the uncorrected gossip strategy.

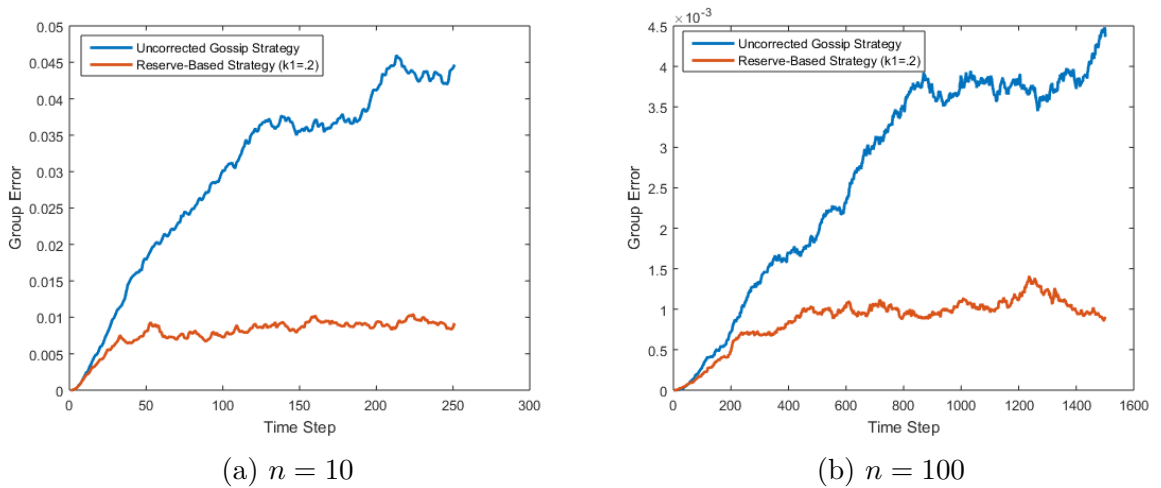


Figure 4.2: Evolution of the group error in an open system for a gossip rate of .9 considering different sizes of system. k_2 is set to 0 and $k_1 = .2$ are considered.

Nevertheless, when the system is large or when the gossip rate is low, the uncorrected

gossip strategy leads still to lower errors. There is no need to strive to use a last message since the group error is small and the little disagreement that is included by the inclusion of the message requires too much time to vanish. This can be pictured by the following example.

Example 1: The mean of a large system is equal to 0 (realist given the distribution). At a certain time, once the system has reach a small disagreement, an agent leaves the system and sends a $\Delta_l = 1$ to an agent that has a good approximation of the mean of the system and a null reserve. The next gossip step in which this agent is included occurs with an agent that has a null reserve and a good approximation of the system and a null reserve. After the gossip step (considering $k_1 = .2$), both approximations are equal to .1 and their reserve is equal to .4. Given the large size of the system, the reserves do not represent a problem against the group error since it has been seen to be small. However, the mean of the system being nearly constant (large system and normal distribution of incoming values) and those two agents having to wait possibly a long time before gossiping again, the disagreement due to the inclusion of this reserve is staying in the system for a while whereas the uncorrected gossip strategy would avoid it. ■

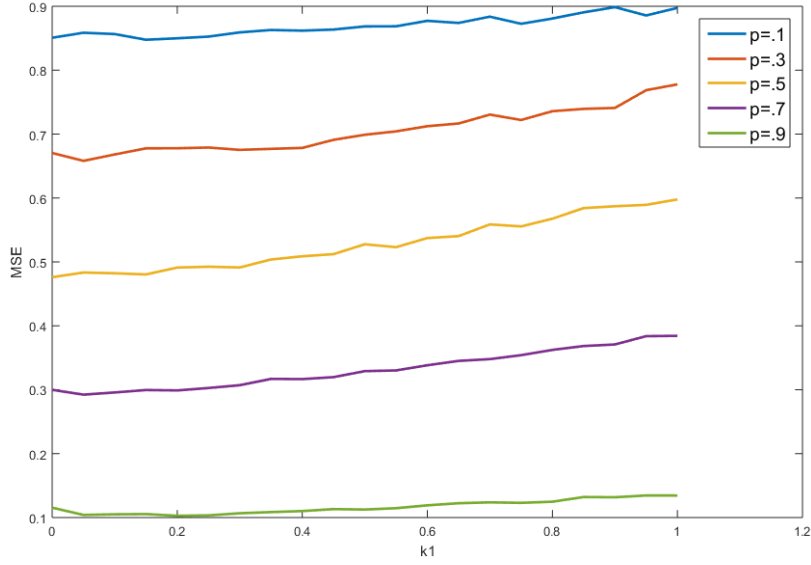
4.2.2 Analysis of the Asymptotic Errors: Influence of k_1

Let us now consider the asymptotic errors resulting from the choice of k_1 . It will give us an indication about which value to use for the analysis of coefficient k_2 . The results are represented on figures 4.3 and 4.4 for systems composed of 20 and 200 agents considering several gossip rates. Once again, a null k_2 is used in order to avoid any immediate *unpleasantness* due to the inclusion of the last message to an the approximation of an agent during a replacement step.

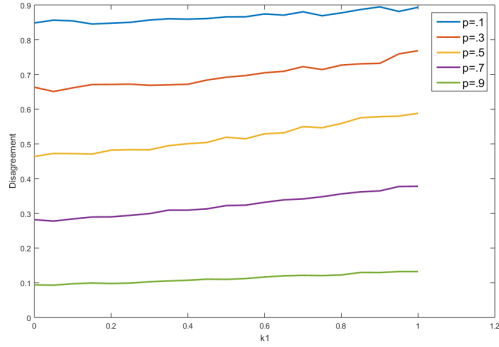
Let us begin this analysis by interpreting the asymptotic group error in both systems. We can see on the corresponding graphs that, in both cases, for high probabilities of gossip, the curves decrease as k_1 raises while for low probabilities, the opposite effect happens. It comes from the fact that, if the agents gossip a lot, they are never able to include the reserves back in the approximations of the system if $k_1 = 0$. In that case, the only way to make the group error evolve is thanks to a replacement where an agent sends its $\Delta - value$ that will also be stored. Therefore, for large probabilities of gossip, large values of k_1 are required to decrease the group error during the gossip steps.

As already mentioned, low probabilities of gossip imply more replacements that may balance the reserves of the agents in the system. The values of k_1 therefore impact more weakly the evolution of the group error since the gossip steps, and therefore this coefficient, take less importance. It comes also that, including the reserves in the system is not advantageous since it *disturbs* the job already made by the replacements and, given the low number of gossips, they are not likely to get close to a zero group error between two replacements anyway. It comes therefore that, for low k_1 , the group error gets larger as the probability of gossip increases. Eventually, we can still observe that this error is in general really small compared to the disagreement term which explains the strong similitude between the results corresponding to this later term with the total MSE.

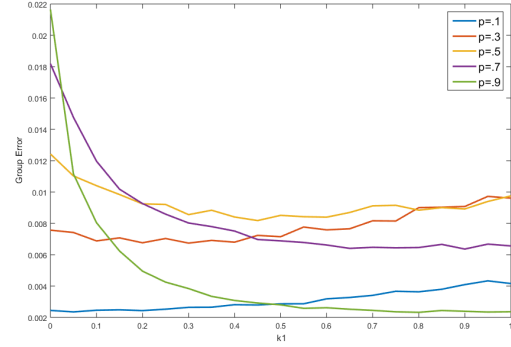
As expected, the asymptotic disagreement increases with k_1 , whatever p_{gos} . It means that the sudden or slow inclusion of the last message through the network does not allow the



(a) Mean Square Error



(b) Disagreement

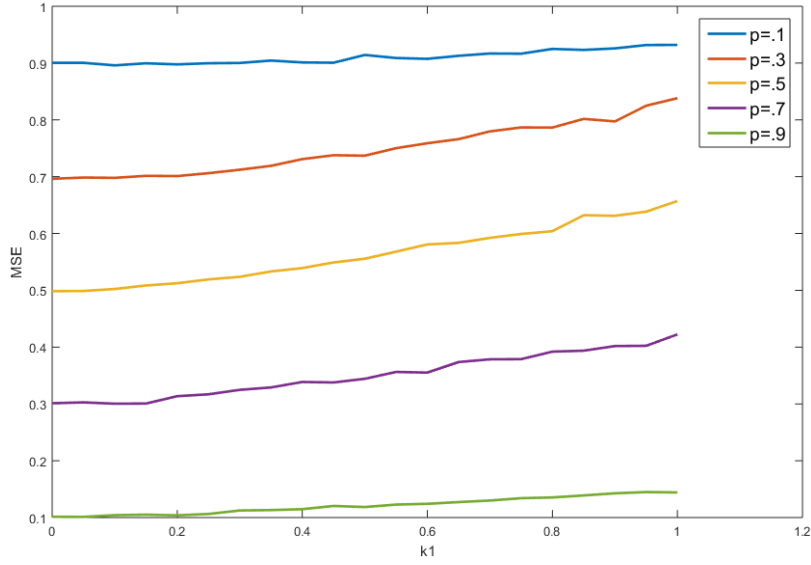


(c) Group Error

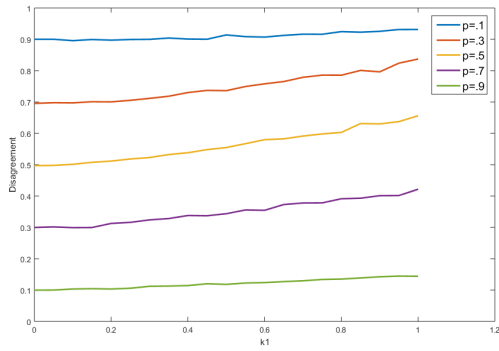
Figure 4.3: Asymptotic error of the MSE, disagreement and group error in the reserve strategy function of coefficient k_1 for different probabilities of gossip and 20 agents ($k_2 = 0$).

agents to reach a consensus more quickly. We have indeed seen that the last message is a source of disagreement. The reserves are actually only a way to move the problem as they work as *hidden errors* that revealed themselves during the gossip steps. The reserves being sent to a random agent in the system, are not values that may help them to reach a better approximation of the system for sure. Therefore, the more the reserves are included, the more the agents may be misleading and therefore get further from an agreement. Also, this disagreement term is higher for larger systems (see example 1).

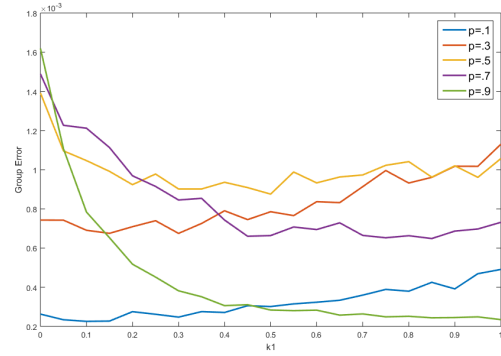
We observe that the curves of the asymptotic MSE are in general increasing, meaning that the uncorrected strategy ($k_1 = 0$ if $k_2 = 0$) is usually more appropriate than using reserves. However, we see on the graph corresponding to a system composed with 20 agents that for high probabilities of gossip, the group error may decrease enough so that it may be interesting to use a non zero value for this coefficient. We can see indeed that the mean square error finds a minimum for $k_1 = .2$ for $p_{gos} = .9$. For $p_{gos} = .7$, the corresponding curve is also initially decreasing. It means that for high probabilities of gossip, the group



(a) Mean Square Error



(b) Disagreement



(c) Group Error

Figure 4.4: Asymptotic error of the MSE, disagreement and group error in the reserve strategy function of coefficient k_1 for different probabilities of gossip and 200 agents ($k_2 = 0$).

error may decrease fast enough to compensate the increase of the disagreement which leads to an optimal strategy different than $k_1 = 0$ if $k_2 = 0$. We have therefore already a hint to answer the question: yes there are cases where using reserves may lead to better results than the uncorrected gossip strategies. This does not seem to be true anymore when the system grows.

4.2.3 Analysis of the Process: Influence of k_2

Given the results of previous sections, we consider here a null value for coefficient k_1 since it leads to the smallest errors in most situations. The evolution through time for different sized systems are represented on figure 4.5 for a gossip rate of .9. Once again, the comments that can be made concerning a lower probability of gossip are similar and will not be given here.

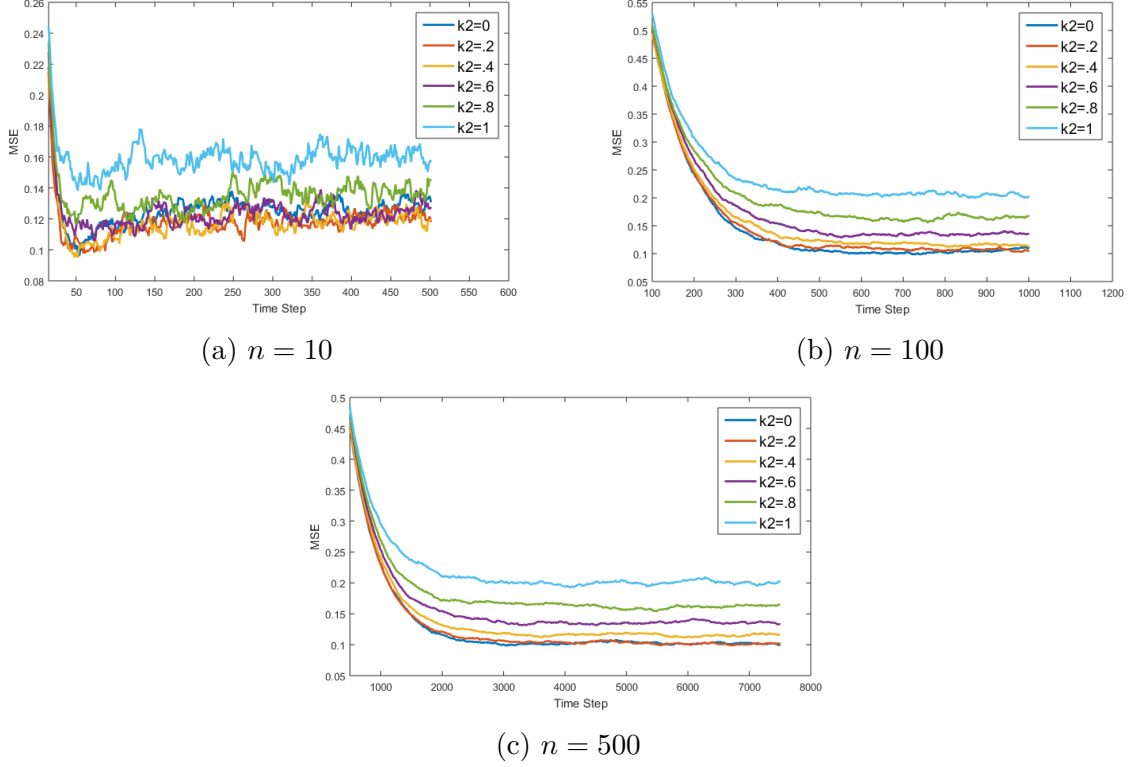


Figure 4.5: Evolution of the MSE in an open system for a gossip rate of .9 considering different sizes of system. k_1 is set to 0 and several values for k_2 are considered.

On this figure, we observe as expected the bad results corresponding to large values for coefficient k_2 , meaning that an important proportion of the last message is immediately included in the system during the replacement steps. This phenomenon is even more apparent for large systems. Indeed, increasing the size of the system up to a certain point results in an important evolution of the curves corresponding to the different values of k_2 . See for example the asymptotic difference between the curves corresponding to $k_2 = 0$ ("lower bound") and $k_2 = 1$. This difference is approximately equal to .04 for a system composed of 10 agents while this difference gets around .1 if the system grows up to 100 agents. Note that a similar comment can be made about the gossip rate that increases the range of asymptotic errors when it decreases.

While $k_2 = 1$ is clearly the worst value for this coefficient, not including at all the last message does not lead to errors significantly lower than using a small part of it. Indeed, $k_1 = .2$ leads to similar errors, whatever the values of the parameters. For small systems and large gossip rates, values for k_2 up to .4 may even be appropriate.

It comes that once again, even if the reserve-based strategy can approach the results of the uncorrected gossip strategy, it does not seem to provide lower errors. However, if it seems to be an improvement of the last message strategy, the good behaviour of the reserve-based strategy for low coefficients of reinsertion of the reserves/last messages loses the interest to get a zero group error (while avoiding large disagreement due to those inclusions).

For small systems where the gossip rate is important, small, non zero, coefficients might give better results than the uncorrected gossip strategy. It does however not seem to be true for all gossip rates and sizes of system.

Note finally that k_2 seems to have a larger influence on the error than k_1 . Observe for example that, for 500 agents in the system, the difference between the asymptotic errors corresponding to $k_1 = 0$ and $k_1 = 1$ (considering $k_2 = 0$) is close to .05 while the same difference is close to .1 when $p_{gos} = .9$. This phenomenon comes from the fact that the unwanted effect of the last message is immediately appearing as the message is totally included during the replacement steps, while it is attenuated in the second one by its share if it is included during the gossip steps. In other words, in the former case, there is no group error but large disagreement. In the latter, a group error, that appears to be most time insignificant, arises but the disagreement that comes from the inclusion of the reserves is weaker since the message is then shared between two agents¹.

4.2.4 Analysis of the Asymptotic Errors: Influence of k_2

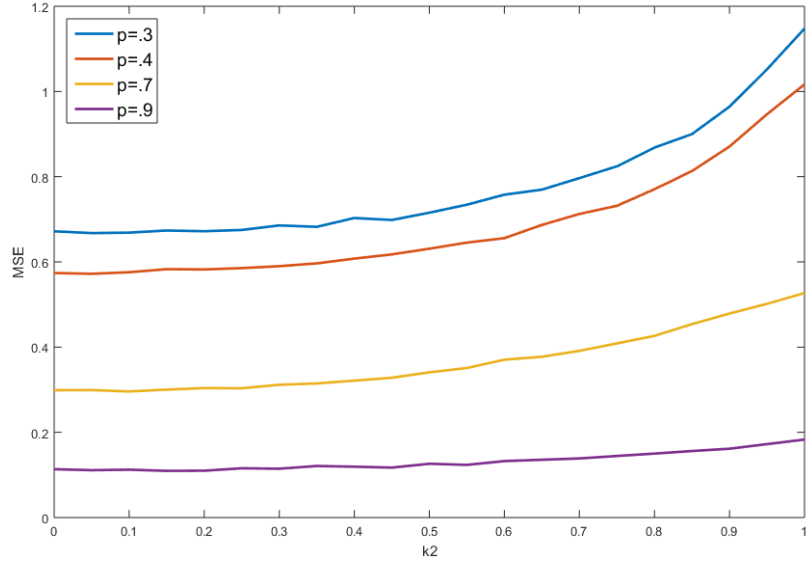
Let us consider the influence on the asymptotic errors of the coefficient linked with the immediate reinsertion of the $\Delta - values$ in the replacement steps. We consider again the case where $k_1 = 0$. These results are represented on figures 4.6 and 4.7 for system composed of respectively 20 and 200 agents.

On the contrary to the results of the k_1 -dependence, the asymptotic group error decreases as k_2 increases, whatever the probability of gossip. Indeed, this coefficient is associated with an immediate insertion of the last message. It comes that, whatever the size of the system, large gossip rates imply that the group error is not frequently *corrected* by a replacement, and it is therefore useful to include a large part of the last message into an agent's approximation so that the evolution of the group error is moderate as a replacement occurs. When the value of k_2 is small, the group error is weaker for small gossip rates. In that case, the group error evolves frequently and is therefore compensating itself. However, as k_2 increases, the reserve-based strategy gets closer to the last message algorithm that has been shown to have no such error. Moreover the closer it gets to 1, the more the last message is reincluded and therefore the less group error is introduced in the system. This term is therefore now equal to 0 when $k_2 = 1$ (exactly the last message strategy, whatever k_1). Keep in mind that k_1 allows to decrease the group error during the gossips steps while k_2 avoids to include such error in the system.

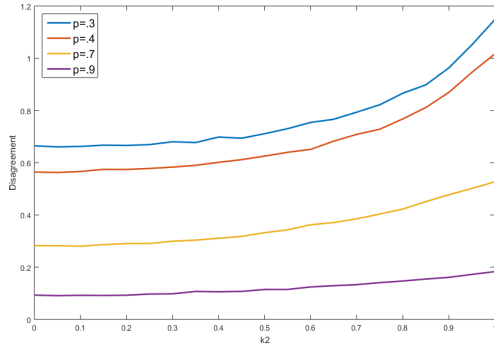
Once again, the group error due to the values in reserve is really low. The curves of this term of error function of k_2 decreases faster when the probabilities of gossip is high. Indeed, when the gossip rate is small, the group error is frequently corrected thanks to the leaving $\Delta - values$. Therefore, the coefficient k_2 has a lower influence in that case than if the gossip rate is large. In that case, the group error is rarely evolving and it is therefore interesting to deal with it immediately, and therefore to increase k_2 .

Note finally that the comment made about the influence of the size of the system and the gossip rate on the asymptotic errors is observed here. We see indeed on those figures that, the smaller the gossip rate, the more important the MSE increases with k_2 . Also, the larger the system the larger the asymptotic MSE increases with k_2 .

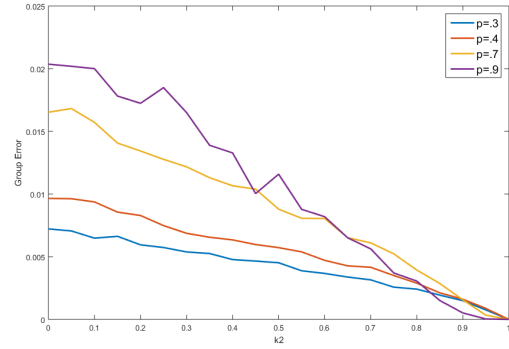
¹Note that both agents may have a reserve to share in that case. Their inclusion may therefore decrease the disagreement due to last messages if they annihilate each other (if the reserves have opposite signs). If they do not, that will lead to a gossip similar than the one in the last message strategy, except that a possible large disagreement due to the time spent by the last message in the approximation of an agent is avoided thanks the reserves.



(a) Mean Square Error



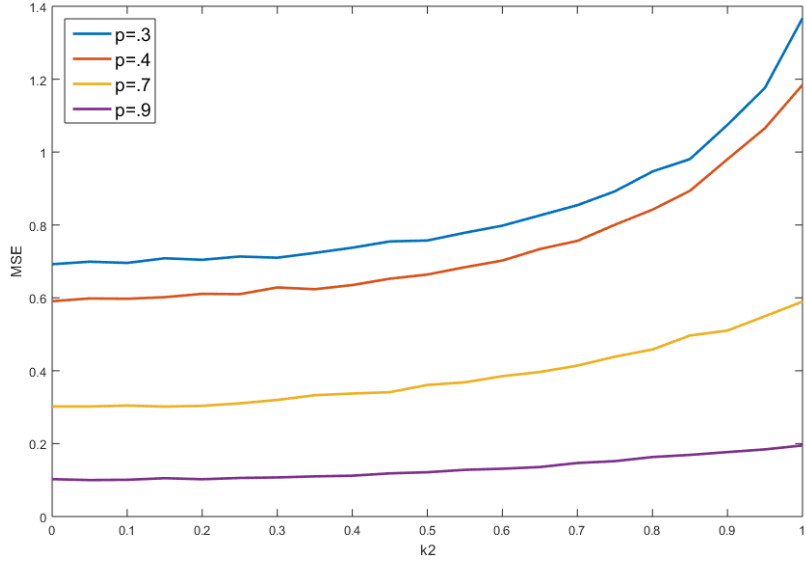
(b) Disagreement



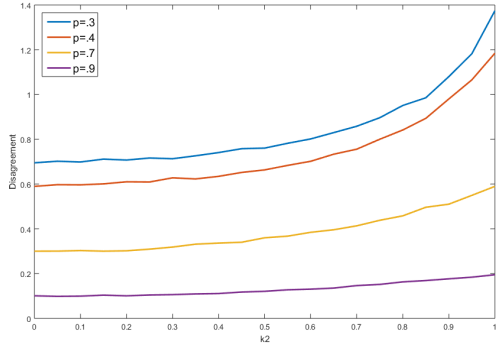
(c) Group Error

Figure 4.6: Asymptotic error of the MSE, disagreement and group error in the reserve strategy function of coefficient k_2 for different probabilities of gossip and 20 agents ($k_1 = 0$).

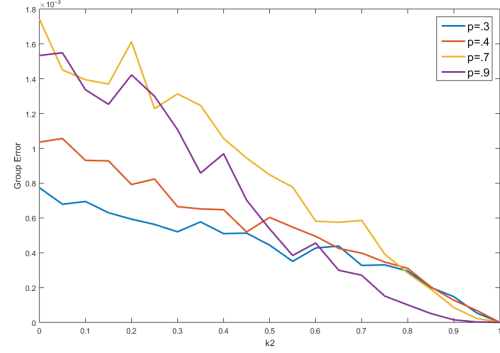
We have seen that, even if using reserves might be interesting for small systems where the gossip rate is large, this reserve-based algorithm, that seems to be an interesting update of the last message strategy, does not lead to desired results. Indeed, the group error remains small (for large enough systems) in any case, whether the reserves are quickly included or not. However, including the last messages (or the reserves) in the system becomes quickly a source of large disagreement, even for moderate values of the coefficients. One final comment that we can make in that situation is that, if the system is likely to close for a long period, it might be useful to use non-zero coefficient k_1 so that the agents may reach a null group error for sure.



(a) Mean Square Error



(b) Disagreement



(c) Group Error

Figure 4.7: Asymptotic error of the MSE, disagreement and group error in the reserve strategy function of coefficient k_2 for different probabilities of gossip and 200 ($k_1 = 0$).

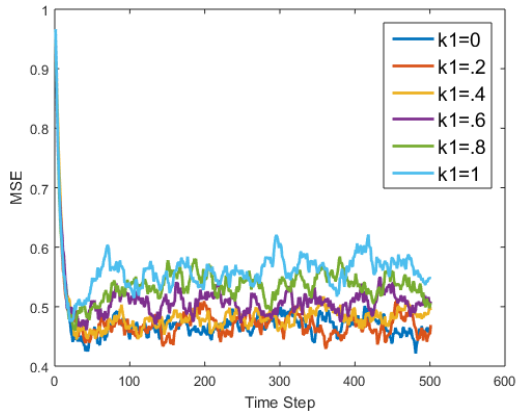
4.3 Time-Depending Distribution of the Incoming Values

Let us now consider the case where the mean of the distribution increases through time. As we have seen that both initial strategies are weakly influenced by a slow increase of the mean of the distribution of the incoming values, we will here consider the case where $\mu(t)$ increases with a high rate in order to observe the effect of the coefficients in the case of evolution distribution. In that case, we will compare different values for both parameters while the other is fixed to zero, given the results of previous section.

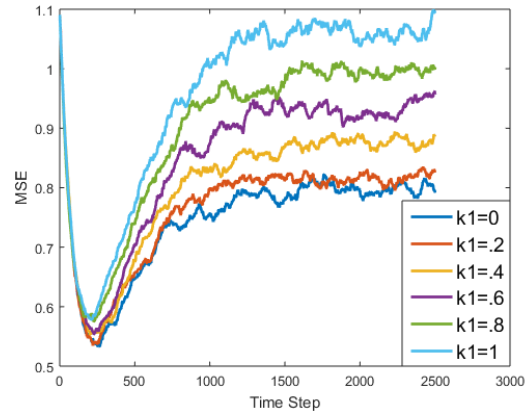
4.3.1 Influence of k_1 and k_2

Figures 4.8 and 4.9 represent the evolution of a mean process given several values for respectively k_1 and k_2 , the other one being fixed to zero. In those results, μ_{max} is equal to 10. Several values are considered for the gossip rate and for the number of agents in the system.

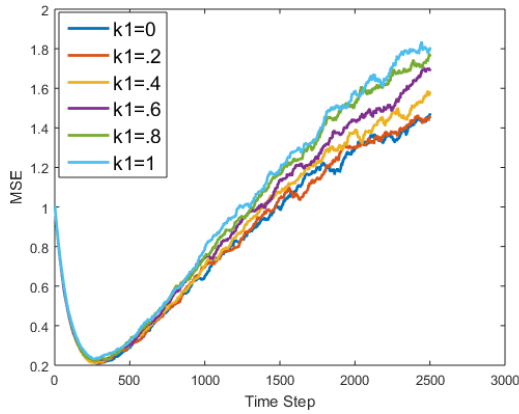
Given the similarities between the results in that section and those in the previous one, we will not extend the comments concerning that situation. Those results indeed raise the same characteristics according to the coefficients such that, the uncorrected gossip strategy still seems to be a lower bound for the reserve-based algorithm and the last message algorithm an upper bound. The evolution of these two bounds for $\mu(t)$ strongly increasing through time has already been analyzed in previous chapter. The intermediate results can be drawn from the results stated in the previous section associated to those given in section 3 of the previous chapter.



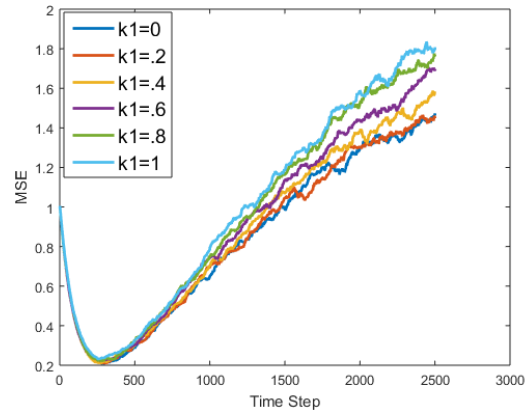
(a) $p_{gos} = .6, n = 10$



(b) $p_{gos} = .6, n = 100$



(c) $p_{gos} = .9, n = 10$



(d) $p_{gos} = .9, n = 100$

Figure 4.8: Mean square error for different values of coefficients k_1 ($\mu(t)$ increases linearly up to 10).

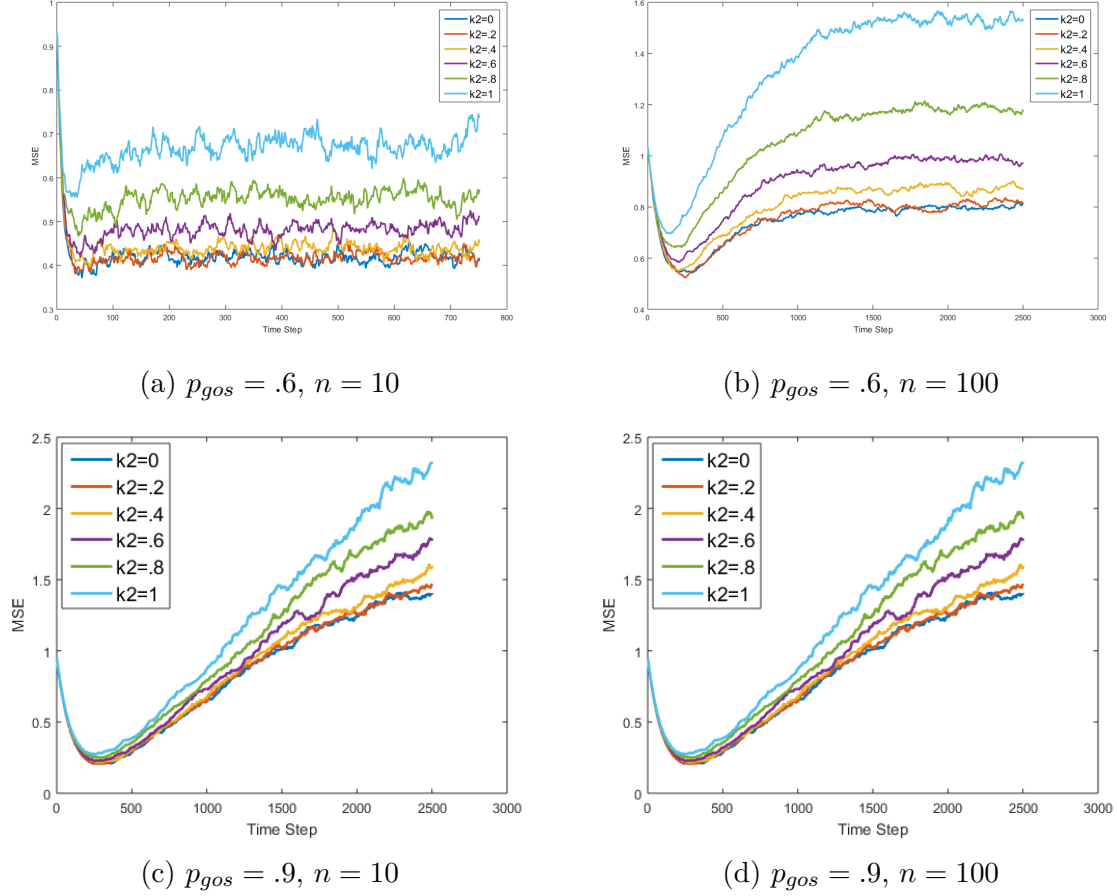


Figure 4.9: Mean square error for different values of coefficients k_2 ($\mu(t)$ increases linearly up to 10).

4.4 Conclusion

The goal of this chapter was to design a strategy that slowly includes the last message in the system in order to avoid the large disagreement due to its sudden insertion during a replacement step. The reserve-based strategy therefore imposed the messages to be first stored as reserves by the agents that receive them, according to a proportion k_2 . They are then included in the systems during the gossip steps given a coefficient k_1 .

We have seen in this chapter that the results of this strategy are not conclusive. We have indeed observed the effect for several values of those coefficients. The corresponding results mostly belong to an interval where the bounds are the results coming from the uncorrected gossip and last message strategies. Two main information come from there. On the one hand, small values for the coefficients lead to results close to the uncorrected gossip strategy. In that case, and of course considering nonzero value for k_1 , the agents may reach a null group error between two replacements if they are able to gossip enough. This represents an advantage compared with the uncorrected gossip strategy. This advantage is of course weaker for large systems where the group error is close to zero anyway in the uncorrected gossip strategy. On the other hand, when the coefficients increase, they are not only doing a larger disagreement as the messages or reserves are included in the system, but they are also source of group error that may remains for a long time in the

system if it is large for example, as long as $k_2 \neq 1$.

This reserve-based algorithm is therefore an improvement of the last message strategy given it decreases its error, but does now present group error most of the time, which was a behaviour that was desired to be avoided. It does nevertheless not lead to better results than the uncorrected gossip strategy in large systems. In conclusion, this new strategy, is a compromise between both initial methods. It presents therefore neither the null group error of one nor the low disagreement of the other. It presents however the possibility, by choosing the coefficients, to either decrease the group error of the uncorrected gossip strategy or decrease the disagreement of the last message strategy.

Chapter 5

Design of a New Strategy

So far, we have seen that neither the last message strategy nor the use of reserves offer the good enough results to get improve those of the uncorrected gossip strategy. Therefore it seems worthless to try hard to design an algorithm that conserves the mean of the values inside the system. This is due on the one hand to the group error that is actually not a real problem in large systems. On the other hand, the large disagreement due to the replacement steps is a main drawback of the last message algorithm.

In this chapter, despite the fact that the last message seemed to be a loss of time, we will keep on trying to design a strategy that uses reserves so that the average of the values in the system remains in it.

On the contrary with previous chapter, the last messages should be treated with more attention. Indeed, so far, the replacement steps were systematic: in the uncorrected gossip strategy ignored the messages, the last message strategy included them immediately and the reserve-based algorithm used them given a fixed coefficient k_2 . Here, we will try to define a strategy where the agents that receive the messages take into account the most information they can when a replacement occurs in order to make a decision about the update of their approximation. In this chapter, those agents take into account the Δ - *values* of the agents sending them as well as both agents own approximations.

In order to design such a strategy adapted to solve the problem where agents in the system receive the messages, we will advance step by step. We will first consider several scenarios that will outline the main drawbacks of both initial strategies. We will then imagine the ideal scenario where the leaving agents are able to interact with those replacing them. In that case, we will consider the two scenarios where the agents reach a consensus for sure or not. We will finally extend it to the problem that we are interested in where the leaving agents do not interact with the one they are replaced by.

Finally, note that we will try to design a strategy where agents decide whether to store or not (and in which proportion) the message they receive based on several information they get about the system. This strategy is therefore design to solve problems where the distribution of the incoming agent is constant as well as those where the distribution is evolving through time.

5.1 Replacements in the Uncorrected Gossip Algorithm and in the Last Message Algorithm

Let us first consider simple scenarios in order to outline some typical characteristics of the outcome of a replacement step in both initial strategies. Given that the disagreement is usually so large compared with the group error, we will try to understand in those first scenarios how to decrease it as the replacements occur. We have indeed seen that this event is usually linked with a large disagreement, mostly in the last message strategy.

The easiest way to analyze the influence of a replacement is by considering small systems so that we can observe each value's evolution through a short period. We will therefore consider in this first section systems composed of 3 agents.

Scenario 1: Let us first consider a system that initially contains 3 agents so that the values in the system are $r_0 = \{0, 0, 0\}$. Let us assume that only the third agent may be replaced (agents 1 and 2 stay in the system) and its substitute always holds the same value. In order to observe as well as possible the outcome of that first case, we consider that the incoming value is far from the two others. We therefore consider that the incoming values are equal to 9. Also, we assume that the replacements are far enough from each other so that agents have the same approximations of the system when a new one happens (agreement found between two replacements). The different states when a replacement occurs are the following¹.

$$\begin{array}{ll}
 \mathbf{r}_0 = \{0, 0, 0\} & \mathbf{x}_0 = \{0, 0, 0\} \\
 \mathbf{r}_1 = \{0, 0, 9\} & \{0, 0, 9\} \rightarrow \mathbf{x}_1 = \{3, 3, 3\} \\
 \mathbf{r}_2 = \{0, 0, 9\} & \{3, 3, 9\} \rightarrow \mathbf{x}_2 = \{5, 5, 5\} \\
 \mathbf{r}_3 = \{0, 0, 9\} & \{5, 5, 9\} \rightarrow \mathbf{x}_3 = \{6.33, 6.33, 6.33\} \\
 \mathbf{r}_4 = \{0, 0, 9\} & \{6.33, 6.33, 9\} \rightarrow \mathbf{x}_4 = \{7.22, 7.22, 7.22\} \\
 & \vdots \\
 \mathbf{r}_\infty = \{0, 0, 9\} & \mathbf{x}_\infty = \{9, 9, 9\}
 \end{array}$$

The last expression comes from the fact that $\bar{x}' = \bar{x} + \frac{1}{3}(9 - \bar{x}_3) = \frac{2}{3}\bar{x} + 3$ (given that an agreement is supposed to be found between two replacements). Previous equation is the expression of a sum of a geometric series with common ratio $\frac{2}{3}$ whose first term equal to 3. It leads therefore to

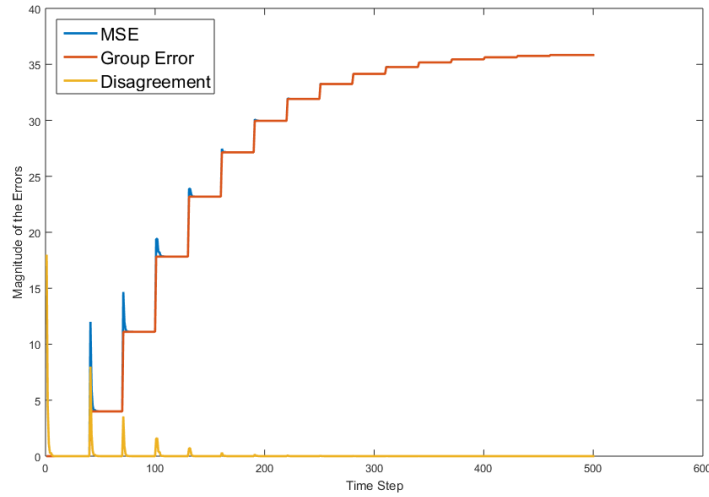
$$\lim_{t \rightarrow \infty} \bar{x}_t = 3 + 2 + 1.33 + \dots = \lim_{t \rightarrow \infty} 3 \cdot \frac{1 - \frac{2}{3}^t}{1 - \frac{2}{3}} = 9.$$

It comes that $\bar{x}_\infty = 9 \gg 3 = \bar{r}_\infty$. In other words, we have just shown that, in such scenario, the uncorrected gossip strategy leads to really bad results since, even if the disagreement tends to zero (the new agents have a value close to the approximations in the system as time goes on) but the group error tends to² 36. ■

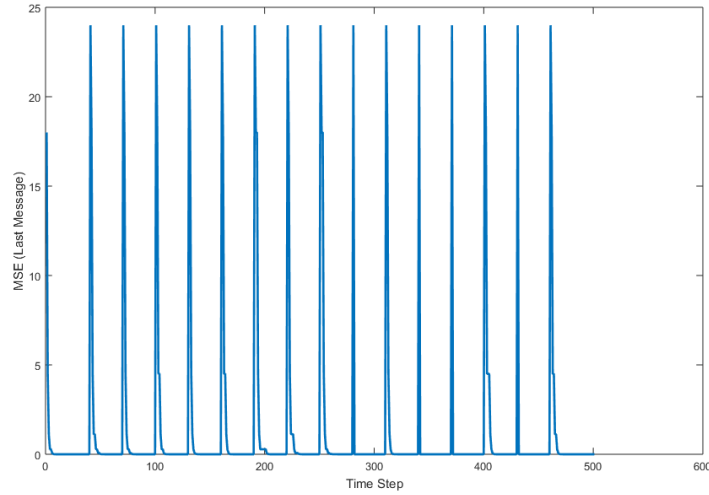
¹Those results are directly linked with the proves in chapter 2.

²The group error is the square of the difference between the true mean of the system and the mean of the approximation of the agents composing it (here $(3 - 9)^2$).

The behaviour of both strategies in that first scenario is represented on figure 5.1. We can observe, as mentioned above, that the group error increases until it reaches 36. Initially, the group error is null and the disagreement is large (in those graphs, $r_0 = \{0, 0, 9\}$). Between two replacements, the group error remains constant and the disagreement decreases until it becomes null (consensus reached). The MSE is therefore equal to the group error as a new replacement occurs. During the first replacements, the disagreement curve undergoes important jumps since the incoming value is far from those currently in the system. Little by little, the approximations of the agents tend toward 9 and the disagreement tends therefore to 0. The MSE fits then the curve of the group error for late times.



(a) Error terms in the uncorrected gossip strategy



(b) MSE in the last message strategy

Figure 5.1: Results corresponding to scenario 1 for the uncorrected gossip and the last message algorithms.

Concerning the last message strategy, we can observe the repetitive behavior of the error on the corresponding graph. The peaks, that occur at the replacement steps, are quickly disappearing as the MSE quickly reaches down zero.

Those two graphs represent quite well the main weakness of both strategies. On the one hand, the uncorrected gossip strategy may lead toward a large group error, and on the other hand, the last message strategy is linked to important peaks of the error. However, this latter algorithm enables the agents to quickly find the appropriate consensus.

It comes that neither of those two strategies offer acceptable results. However, if the leaving agents could interact with the incoming one during the replacements, the new agents should include the Δ - *value* into their own approximation. Indeed, given that we assumed that the agents found a consensus before the replacement, the resulting error would be null. Indeed, the outcome of a replacement in that first scenario would be: $\mathbf{r}' = \{0, 0, 9\}$, $\mathbf{x}' = \{3, 3, 3\}$ and $\mathbf{\Delta}' = \{3, 3, -6\}$. In other words, when the new agent takes the place of agent 3, it immediately changes its approximation by $x_e = r_e + \Delta_3 = 9 - 6 = 3$. Therefore, neither a group error nor a disagreement comes from that replacement.

Note also that, in that case, the reserves may be useful if the leaving agents are not allowed to gossip with the incoming ones. Indeed, the group error coming from the reserves is quickly vanishing if the reserves are included back in the system. Therefore, it may be useful to take advantage of the lower disagreement of the uncorrected gossip strategy while conserving a null group error in the system.

This example is also useful to realize that the group error would quickly decrease if agent 1 or 2 gets replaced by an agent holding a value equal to 9. Indeed, the mean of the system will increase from 3 to 6. If the 3 agents are replaced by a value equal to 9, the group error as well as the disagreement will asymptotically decrease to 0. Therefore, the group error that may initially be associated to bad results may eventually become really small.

Let us now skip to another scenario where the system is once again composed of three agents. Its goal is to observe the effect on the error given the replaced value and the incoming one.

Scenario 2: Assume now that the agents of a system hold initially the following values and approximations: $\mathbf{r} = \{-1, 0, 1\}$, $\mathbf{x} = \{1, 0, -1\}$ ($\rightarrow \mathbf{\Delta} = \{2, 0, -2\}$). Consider the following situations in which the leaving agents do not send any last message.

- Agent 1 is replaced by an agent with the same value as its approximation ($x_1^+ = r_1^+ = 1$, where the + sign refers to the incoming agent taking "place 1"). The disagreement has not changed (equal to $\frac{2}{3}$) but the group error has quite largely ($0 \rightarrow \frac{4}{9}$). The total error is therefore $\frac{10}{9}$.
- Agent 1 is replaced by an agent holding a value equal to the mean of the system ($x_1^+ = r_1^+ = 0$). In that case, the group error is the same ($0 \rightarrow \frac{4}{9}$) but the disagreement decreases ($\frac{2}{3} \rightarrow \frac{2}{9}$). The total error is therefore $\frac{2}{3}$.
- Agent 1 is replaced by an agent holding the same value ($x_1^+ = r_1^+ = -1$). In that case, the new group error is still equal to $\frac{4}{9}$ but the disagreement goes again to $\frac{2}{9}$. The total error is therefore $\frac{2}{3}$.
- Agent 1 is replaced by an agent having a value lower than 1 ($x_1^+ = r_1^+ = -.5$). While the group error is still the same as in previous cases, the new disagreement is now $\frac{3}{18}$.
- Agent 1 is replaced by a huge value ($x_1 = r_1 = 100$). The group error is once again the same but the disagreement explodes. Note that, even if the group error is still

the same, it is less significant compared to the mean of the system.

To summarize, the group error is of course not influenced by the incoming agent but depends only on the leaving one (relying on the $\Delta - value$ used in last message). However, the disagreement is strongly influenced by the incoming agent. Indeed, in that scenario where the initial mean is 0, too large values in magnitude will lead to huge disagreements. We will therefore draw advantage of this value to reduce the disagreement included in the system as a replacements occurs.

Consider again, the second case ($r_1^+ = 0$). However, instead of considering a replacement step similar of the type of the uncorrected gossip strategy, agent 1^+ uses a part of Δ_1 into its approximation (we again assume that the new agent interact with the leaving one) so that $x_1^+ = \frac{1}{3}$ and $s_1^+ = \frac{5}{3}$. It comes that the group error of that situation is equal $\frac{25}{81} < \frac{4}{9}$ and the disagreement is $\frac{5}{27} < \frac{2}{9}$. We see therefore that it is possible to use the last message to reduce both the group error and the disagreement linked with a replacement. ■

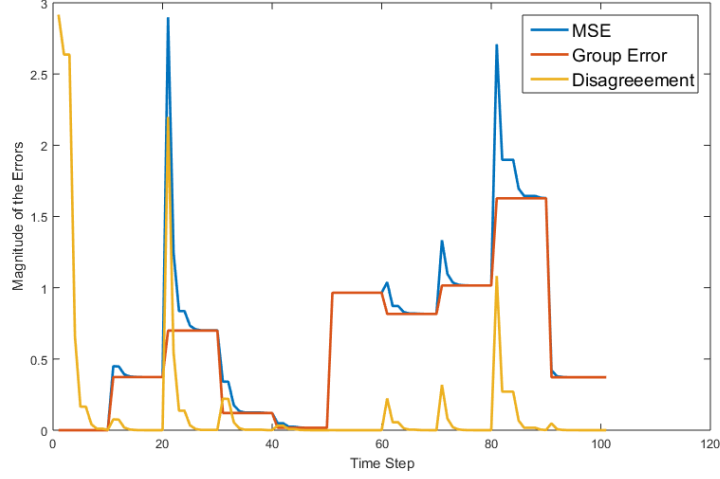
So far, we have imposed the incoming values as well as the leaving agents. It is now time to consider a *random process*. We again impose that the agents reach a consensus between two replacements and the incoming values are again normally distributed. The agents that leave the system, gossip or receive the message are randomly chosen.

Scenario 3: As we have already seen that 3 agents can find a consensus in less than 10 time steps, we impose that a replacement occurs each 10 time steps. We consider here that the distribution of the incoming values evolves through time and that $\mu(t)$ increases from 0 to 10 over 100 time steps so that the difference of the mean of the distribution is significant between two consecutive replacements. The different values at the time steps corresponding to the different replacements are given in table 5.1 and the evolution of the error in both strategies is represented on figure 5.2.

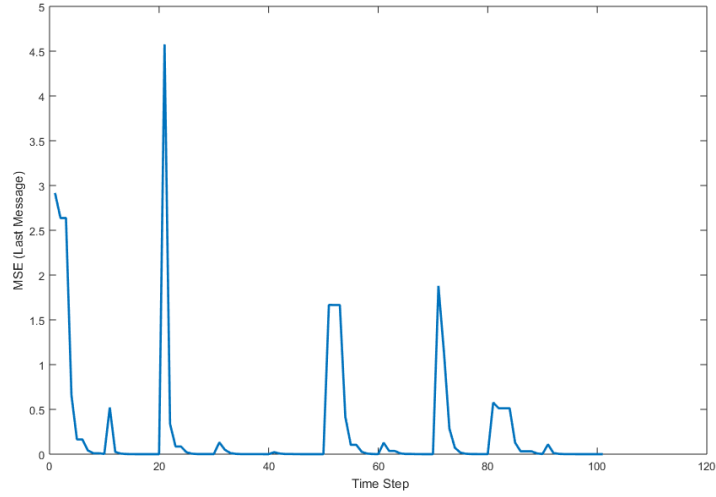
# Repl.	Replaced Agent	Agent 1	Agent 2	Agent3	$\bar{r}$	$\bar{x}$
0		0.54	1.83	-2.26	0.038	0.038
1	1	0.19	1.83	-2.26	-0.08	0.09
2	2	0.19	3.10	-2.26	0.34	1.07
3	3	0.19	3.10	2.33	1.87	1.49
4	3	0.19	3.10	2.94	2.08	1.97
5	2	0.19	6.02	2.94	3.05	3.32
6	2	0.19	7.04	2.94	3.39	4.56
7	1	6.81	7.04	2.94	5.60	5.31
8	1	9.60	7.04	2.94	6.53	6.74
9	1	9.92	7.04	2.94	6.63	7.80

Table 5.1: Results corresponding to the process represented on figure 5.2

Let us first analyze the behavior of the last message strategy. The first replacement (after 10 time steps) is linked with a small variation of the MSE. This can be explained by the fact that, on the one hand, the leaving agent sends a small $\Delta - value$ ($\approx .5$) since its own value was already close to the mean of the system. Therefore the agent receiving the message is not influenced much by this one. On the other hand, the incoming agent brings a value in the system that does not influence too much the mean of the system and has a value close too it. It comes that the error made by:



(a) Different errors related to the uncorrected gossip strategy



(b) Mean square error related to the last message strategy

Figure 5.2: Results corresponding to scenario 3 ($\mu(t)$ evolves linearly between 0 and 10 over 100 time steps).

- the incoming agent is small due to an own value close to the new mean of the system
- the last message is small due to the moderate Δ – value
- the leaving agent is small since the mean did not change much and had therefore a good approximation of this value initially

The same logic may be applied to the replacement happening after 40 time steps. The second replacement is the first linked to an important raise of the MSE. The reason is related to the following facts:

- since agents reach an agreement, the leaving agent having a value far from its approximation before replacement (the mean of the system) sends a large message.

This last message takes the agent receiving it far from the new mean (the mean increases while the last message is negative);

- the incoming value is significantly larger than the leaving one which leads to an important increase of $\bar{r}$. The agent's value that is not influenced by the replacement is therefore taking away from the mean of the system and is therefore linked with a high source of error;
- the incoming value is still far from the new mean. The incoming agent has therefore also a bad idea of the mean of the system.

The errors linked to replacements 3 and 7 are moderate because the messages sent are in the same direction as the evolution of the mean. Indeed, in these two cases, agents with values below the mean get replaced by agent with values above it. Therefore, the mean increases and the agent receiving the message "tends to follow it". In both cases, the incoming values are relatively close to the new mean, the agent receiving the message goes in the same direction as the mean and the last one is still not far away from the new mean since this latest has not drastically changed.

The errors associated to the replacements 6 and 9 (that are the more important among all) may seem surprising since the leaving agents seem to be replaced by values with the same magnitude leading to the mean of the system lowly influenced by the replacement. However, in both cases, the two agents included in the replacement have similar values that are far from the mean of the system, which does therefore not change much. It comes that a large message is sent and the agent receiving it is expelled far away from the mean of the system, since it has already a value way lower than the mean of the system.

Finally, errors made at replacements 5 and 8 are linked with important raises of the mean as the replacements occur. Therefore, the incoming agents are relatively far from the new mean (the new agent brings a value way larger than the leaving agent's one. Since the mean is also influenced by other small values, it increases but is still quite far from the value of the incoming agent). Also the last message is not to neglect and once again goes in opposite directions with the evolution of the mean. Finally, from the increase of the new mean, the last agent is also linked with a quite important error.

In that third scenario, we have outlined that the reserves may be useful to treat such problem. Indeed, the incoming values and the large $\Delta - values$ are linked to the highest errors made during this process. Also, it seems that the agents coming with a naive approximation is also a source of error. Linking the leaving agent with the coming one in such a problem may also be a way to decrease the sudden disagreement. Moreover, if the leaving agents and the incoming ones are able to interact, they could have the information on the evolution of the mean which may be interesting considering replacements 1, 3 and 7. They could therefore make a decision in order to decrease as much as possible the disagreement

Now that we have talked about the last message strategy, let us consider the results of the uncorrected gossip strategy in this situation. More precisely, let us focus on the group error (the disagreement is caused by the magnitude of the incoming agent's value and the way it moves the mean of the system), which is the term toward which the MSE converges between two replacements.

Initially the group error is null. When the first replacement occurs, an agent leaves and is a source of small group error since its own value was close to the initial mean. Therefore, its approximation has weakly been influenced. However, when agent 2 gets replaced for the first time, it creates an important imbalance since its value (1.83) was far from $\bar{x}_1$ (0.09).

But this group error does not increase at each replacement step. Indeed, when agent 3 leaves the system for the first time, it holds a Δ – *value* that is large and its departure decreases the magnitude of the error made by the approximations so far.

As suggested in SCENARIO 1, the highest errors are made (on steps 60 and 90) when the same agent is replaced twice in a row by the similar values, way larger than the current mean of the system. It comes from the fact that it strongly increases the mean of the approximations while weakly impacting the mean of the values in the system. Therefore, the incoming agent has a value closer to the new $\bar{x}$. However, the other agents get far away from it resulting in an exploding Δ for agent 1 in the sixth replacement and for agent 3 in the last one.

Note finally that, even if the values inter-replacements are lower with the last message algorithm, the uncorrected gossip strategy is once again associated with lower peaks when those events occur due to much smaller disagreement raises.

We have now enough data to design our new strategy. One last comment that we can make is that, as the size of the system grows, the increase $\mu(t)$ is less felt by the system. Indeed, if a given value joins a small or large system, it will obviously have less influence on the error given the weight $\frac{1}{n}$ in the error terms. Moreover, the mean of the system will also slower increase.

5.2 Case 1: Message Sent to the Incoming Agent and Consensus Reached

In order to design a strategy meant to solve the general problem, let us first consider the simplify one where the leaving agents interact with the incoming ones and the agents reach a consensus between two replacements. We will little by little adapt this algorithm to get an algorithm designed to solve the global problem. This simplified situation may for example be assimilated to connected captors that may, for a reason or another, reset or change their values. Instead of deleting the information held before reset, the captors store the information until it restarts. We could take as hint that, if this new strategy does not lead to smaller errors than the uncorrected gossip strategy, than this latter algorithm is the best fitted to solve the problem of average seeking.

This first case that we will get interested in is somehow trivial. Indeed, since the agents are able to find an agreement between two replacements and the two agents involved in this step interact together, the incoming agents can behave adequately with the information they received in order to lead to the smallest possible MSE. They can indeed determine a suitable way to find a compromise between the storage in reserves (leading to a group error), or the inclusion of the last message (leading to a possible disagreement). The goal being to make this scenario evolve to a more global problem, we will however not

use direct computation to define the best strategy to adopt at each replacement step. Instead, we will use the reserves to deal with the increase of the disagreement coming from a replacement steps, no matter of the group error.

Let us now present the different steps of this alternative algorithm. A gossip step occurs as in the reserve-based algorithm presented in the previous chapter: the agents that take part in the gossip take as approximation the mean of their current approximation as well as a given proportion of the mean of their reserves (which they also share). We can remind that gossip step here. Note that in the simulations presented in this section, we consider $k_1 = 1$ so that the group error remains the shortest time in the system and leads the agent to a null group error between two replacements.

Algorithm 6 Alternative Strategy I (Gossip)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
 Select uniformly and randomly two agents $i, j \in \mathcal{S}(t)$
 $s_i(t) = s_j(t) = \frac{s_i(t) + s_j(t)}{2}$
 $x_i(t+1) = x_j(t+1) = \frac{x_i(t) + x_j(t)}{2} + k_1 \frac{s_i(t) + s_j(t)}{2}$
 $s_i(t+1) = s_j(t+1) = (1 - k_1) \frac{s_i(t) + s_j(t)}{2}$

In that section, given that a consensus is reached, we suppose that the reserves are null when a replacement occurs. Keep in mind that the goal of this section is to decrease the disagreement linked to a replacement, and therefore to lead the first approximation of the incoming agent as close as possible to the previous mean of the system, which was the value of the last consensus. In that case a group error may be coming from the use of reserve but the initial disagreement is kept as small since the first approximation of the incoming agent tends to the approximations of the agents in the system. A replacement may be characterized by three distinct situations:

1. $r_l \approx r_e \Rightarrow \bar{r}' \approx \bar{r}$
 - since an agreement is supposed to be found (and the group error is null), if the incoming agent has a value close to the leaving one, he gets the Δ - value of the leaving agent (see scenario 1) and therefore the state of the system after replacement is close to its state before it (hence low disagreement and null group error). Therefore $x_e = r_e + \Delta_l \approx \bar{r}'$.

Algorithm 7 Alternative Strategy I (Replacement 1.)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
 incoming agent e , leaving agent l
 $\Delta_l = x_l(t) - r_l$
 $x_e(t+1) = r_e + \Delta_l, s_e(t+1) = 0$
 New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

2. $r_l > r_e$
 - if $\Delta_l \approx 0 \rightarrow$ low influence on the incoming approximation $\rightarrow x_e = r_e + \Delta_l \approx r_e$: nearly equivalent to a replacement step of the uncorrected gossip strategy.

- **if** $\Delta_l > 0 \rightarrow \Delta_l$ leads the new approximation toward the previous mean of the system³ (The approximation with the inclusion of Δ_l is the closest it might be to the previous consensus 0 and the Δ - *value* is positive) $\rightarrow$ Inclusion of Δ_l in the incoming approximation (no group error and lowest possible disagreement)
- **if** $\Delta_l < 0 \rightarrow$ the Δ_l should be carefully treated. Indeed, if $r_e < \bar{r}$, the negative message may only take the agent away from a consensus and should therefore not be used. If $r_e > \bar{r}$, Δ_l should be used only to reach $\bar{r}$. A decision considering how to include Δ_l in this last case needs therefore only to be made when $|r_l - r_e| < |\Delta_l|$ **(a)**.

Algorithm 8 Alternative Strategy I (Replacement **2**.)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
incoming agent e , leaving agent l
 $\Delta_l = x_l(t) - r_l$
if $\Delta_l \approx 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else if $\Delta_l > 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else
 $a = \frac{\Delta_l}{r_e - r_l}$
 $k = \frac{a-1}{a}$
if $k > 0$ **then**
 $x_e(t+1) = r_e + k\Delta_l, s_e(t+1) = (1-k)\Delta_l$
else
 $x_e(t+1) = r_e, s_e(t+1) = \Delta_l$
end if
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

3. $r_l < r_e$

(This last scenario is similar to the previous one).

³ $\bar{r} - x_e = \bar{r} - (r_e + \Delta_l) = \bar{r} - r_e + r_l - \bar{r} = r_l - r_e > 0$

Algorithm 9 Alternative Strategy I (Replacement **3.**)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
if $\Delta_l \approx 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else if $\Delta_l < 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else
 $a = \frac{\Delta_l}{r_e - r_l}$
 $k = \frac{a-1}{a}$
if $k > 0$ **then**
 $x_e(t+1) = r_e + k\Delta_l, s_e(t+1) = (1-k)\Delta_l$
else
 $x_e(t+1) = r_e, s_e(t+1) = \Delta_l$
end if
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

There remains to explain where the decision about case **a**) comes from. Consider the following scenarios where the value of Δ_l is expressed in function of the two values held by the interacting agents as follows:

$$\begin{array}{lll}
 \Delta_l = r_e - r_l & \Delta_l = 2(r_e - r_l) & \Delta_l = 4(r_e - r_l) \\
 \bar{r} = r_e & \bar{r} = r_e + \frac{\Delta_l}{2} & \bar{r} = r_e + \frac{3\Delta_l}{4}
 \end{array}$$

Whatever the scenario, it comes therefore the following general expression:

$$\begin{aligned}
 \Delta_l &= a(r_e - r_l) \\
 \bar{r} &= r_e + \frac{a-1}{a}\Delta_l
 \end{aligned}$$

It comes that we found the appropriate proportion of the message received to include to the incoming agents approximations in order to decrease as much as possible the (usually large) disagreement linked with a replacement. Actually, this leads the value of the agent the closest to the previous average of the values in the systems, keeping a low disagreement as the replacement occurs.

However, this strategy may be linked with an error that comes from the use of the reserves. Also, for the interest of the general problem, this coefficient may not be lower than 0 (message sent in the opposite direction of Δ_l). This choice of coefficient for the inclusion of the reserve is never larger than 1 (message larger than the Δ_l).

Let us now observe the results of this new strategy. In the next sections of this chapter, we will consider the evolution of the error in both cases where the distribution is constant

or not. It has however here low interest given the small number of replacements. The results corresponding to a constant distribution is represented on figure 5.3.

On that figure, we see that we have finally found a strategy that avoids the large peaks of the results associated to the last message strategy and that leads to a null error before a new replacement occurs. For small systems, this new strategy is even better than the uncorrected gossip strategy. The group error is indeed quickly reaching zero (while the uncorrected gossip strategy is associated with a significant group error) and the disagreements coming from the replacements are mitigated. However, when the system grows, the disagreement is also dealt adequately in that new strategy as replacements occur (lower MSE at the replacements steps). However, the reserves included during the gossip steps leads to a slower decrease of the error of this new strategy than in the other two. The reserves can actually be seen as *hidden error*. It comes that the gossip are likely to decrease more slowly the disagreement.

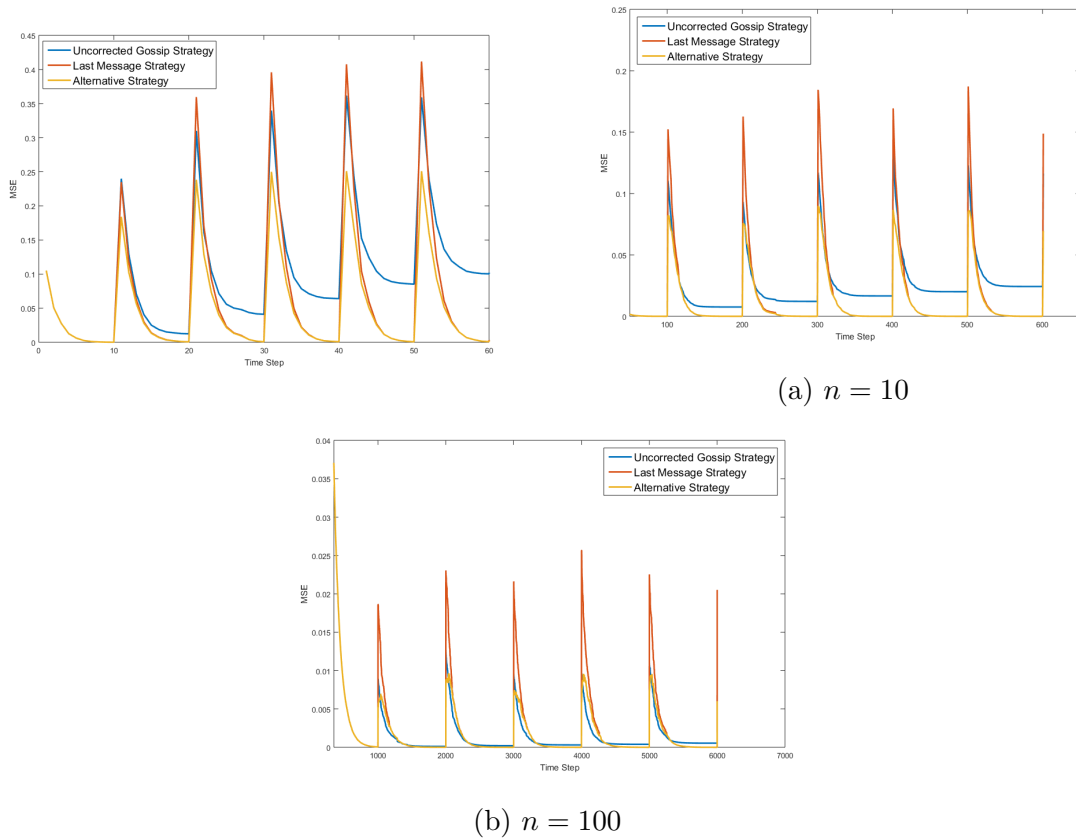


Figure 5.3: Results corresponding to the first case considered in the design of an alternative strategy.

5.3 Case 2: Message Sent to the Incoming Agent and Consensus Not Reached

Given the results in previous section, we have still hope to design an algorithm that would decrease the disagreement as the replacements occur in order to improve the performances

of the uncorrected gossip strategy. Let us therefore go one step further in the conception of such a well adapted algorithm meant to solve the general problem. On that purpose, we now assume that the agents do not find a consensus for sure between two replacements. We will therefore consider several gossip rates as in previous chapters.

In this section, we will consider the same strategy in the gossip step. However, instead of considering a unitary value for k_1 , that has been seen not to be adequate, we will instead use $k_1 = .2$. In previous chapter, we have indeed observed that such coefficient leads to results close to those coming from $k_1 = 0$. Moreover, that value leads the group error to smaller values.

Compared with previous case, the agents are not supposed to reach any consensus between two replacements. The replaced agents may therefore still hold non-zero reserves and may have a bad approximation of the mean of the system. We will however consider here that the agents that enter the system will react as if the values given by the leaving agents were accurate. In that case, the new agents adopt the same strategy so that its first approximation tends to the one of the leaving agent. In other words, it will not try to decrease the disagreement, but instead make it stable as a replacement occurs. During those interactions, the remaining reserves are not taken into account in the initial approximation of the incoming agents and are immediately restored.

Replacements **1.**, **2** and **3.** can finally be rewritten considering those new conditions.

Algorithm 10 Alternative Strategy II (Replacement **1.**)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
incoming agent e , leaving agent l
 $r_e \sim \mathcal{N}(\mu(t), \sigma^2)$
 $\Delta_l = x_l(t) - r_l$
 $x_e(t+1) = r_e + \Delta_l, s_e(t+1) = s_l(t)$
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

The comparison of the two initial strategies with this new one is represented on figure 5.4 for a constant distribution and on figure 5.5 for $\mu(t)$ increasing up to 10. Once again, we can see that this new strategy is quite efficient and improves the results of both the last message and the uncorrected gossip strategy. When the distribution is constant, the error coming from this new strategy is even lower than the one in the uncorrected gossip strategy for small probabilities of gossip or in large systems.

However, the case where the system is large considering a small gossip rate and importantly evolving distribution for the incoming values, remains a situation that should be treated with the uncorrected gossip strategy. An explanation for that behaviour is that, since the agents are not likely to find an agreement in that case, the leaving agents may be associated with a high disagreement. Attempting to get close to their approximation may therefore not decrease this possibly large error.

Algorithm 11 Alternative Strategy II (Replacement **2.**)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
incoming agent e , leaving agent l
 $\Delta_l = x_l(t) - r_l$
if $\Delta_l \approx 0$ **then**
 $s_e(t+1) = s_l(t), x_e(t+1) = r_e + \Delta_l$
else if $\Delta_l > 0$ **then**
 $s_e(t+1) = s_l(t), x_e(t+1) = r_e + \Delta_l$
else
 $a = \frac{\Delta_l}{r_e - r_l}$
 $k = \frac{a-1}{a}$
 if $k > 0$ **then**
 $x_e(t+1) = r_e + k\Delta_l, s_e(t+1) = (1-k)\Delta_l + s_l(t)$
 else
 $x_e(t+1) = r_e, s_e(t+1) = s_l(t) + \Delta_l$
 end if
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

Algorithm 12 Alternative Strategy II (Replacement **3.**)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
if $\Delta_l \approx 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else if $\Delta_l < 0$ **then**
 $s_e(t+1) = 0, x_e(t+1) = r_e + \Delta_l$
else
 $a = \frac{\Delta_l}{r_e - r_l}$
 $k = \frac{a-1}{a}$
 if $k > 0$ **then**
 $x_e(t+1) = r_e + \Delta_l, s_e(t+1) = (1-k)\Delta_l$
 else
 $x_e(t+1) = r_e, s_e(t+1) = \Delta_l$
 end if
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

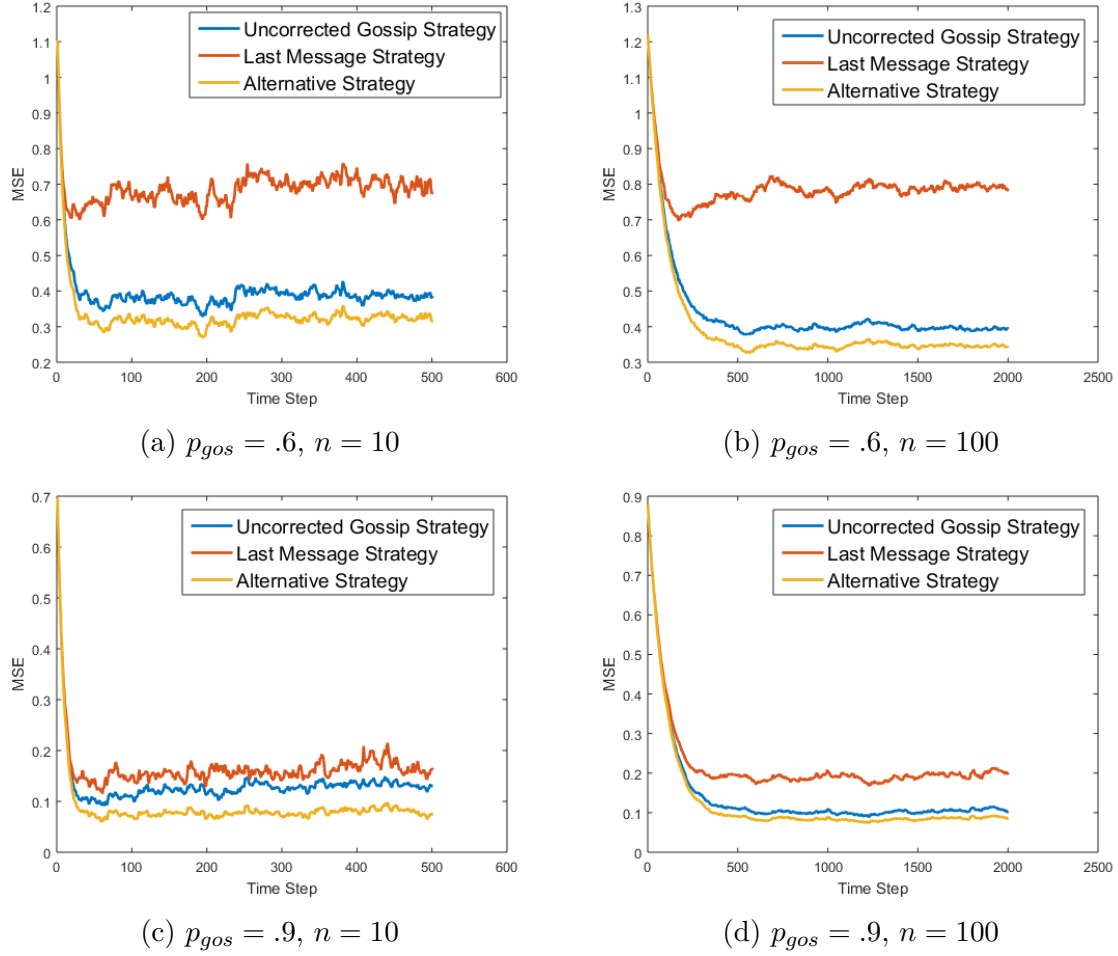


Figure 5.4: Results corresponding to the second case considered in the design of an alternative strategy (constant distribution).

5.4 Case 3: Message Sent to a Random Agent in the System

5.4.1 Alternative Strategy

The results of the previous case are promising and we can now tackle the problem of interest where leaving agents send their message to a random agent in the system. The agents that receive the message ignore the values of the incoming ones. Also the agents making the decisions may also have a new approximation of the system such that $x_m(t) \neq r_m$. The decisions are therefore not taken considering the values held by the leaving agents and the agents that receive the message, but on their approximations instead.

We have seen so far that the uncorrected gossip strategy draws its strength mainly from the fact that only one approximation in the system changes as a replacement occurs. We can therefore take inspiration from that fact to design our new strategy. In this spirit, the goal of the message will be to get the approximation of the agent receiving this message as close as possible to the value of the leaving agent so that both methods become similar in the sense that they tend to change only one value. Indeed, the uncorrected gossip strategy

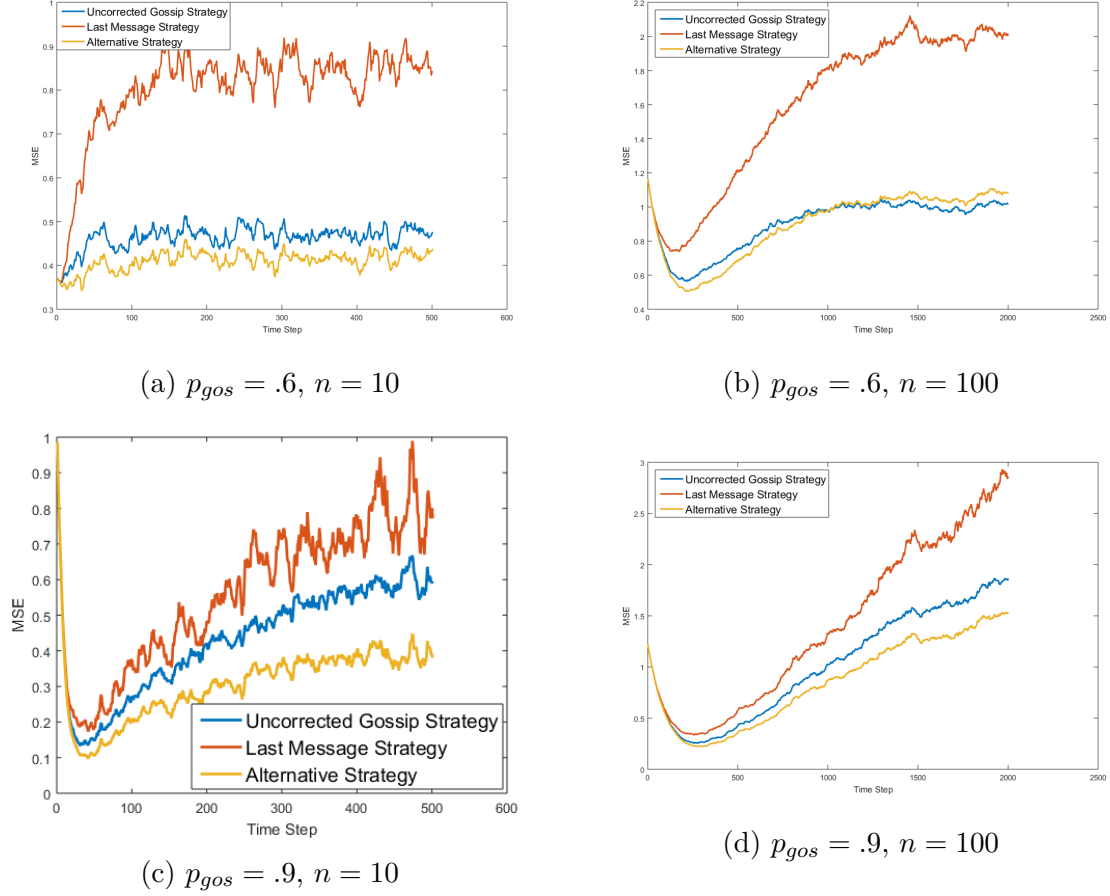


Figure 5.5: Results corresponding to the second case considered in the design of an alternative strategy ($\mu(t)$ increases linearly up to 10).

changes the approximations of the leaving agents by those of the incoming ones. The strategy we will design here tends to replace the approximations of the agents receiving the message by those of the incoming agent, the approximations of the leaving agents being approximated by the agents that receive the messages.

As in previous case, we consider a small, non-negative coefficient for k_1 given the results presented in the previous chapter.

Once again, three cases are considered.

1. $x_m(t) \approx x_l(t)$
 - The approximation of the agent receiving the message is close to the leaving agent's one. An agreement is possibly reached. The agent should therefore not change its approximation.
2. $x_m(t) < x_l(t)$
 - **if** $\Delta_l \approx 0 \rightarrow$ low influence on the approximation $\rightarrow x_m(t+1) = x_m(t) + \Delta_l \approx x_m(t)$ (nearly equivalent to a replacement step of the uncorrected gossip strategy)
 - **if** $\Delta_l > 0 \rightarrow \Delta_l$ leads the approximation of agent m toward $x_l(t)$. The message is included as long as it does not get over $x_l(t)$.

Algorithm 13 Alternative Strategy III (Replacement 1.)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
agent receiving the message m , leaving agent l
 $\Delta_l = x_l(t) - r_l$
 $x_m(t+1) = x_m(t), s_m(t+1) = s_m(t) + s_l(t) + \Delta_l$
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

- if $\Delta_l < 0 \rightarrow \Delta_l$ leads the approximation of agent m away from $x_l(t)$ and the message is therefore stored as reserve.

Algorithm 14 Alternative Strategy III (Replacement 2.)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
agent receiving the message m , leaving agent l
 $\Delta_l = x_l(t) - r_l$
if $\Delta_l \approx 0$ or $\Delta_l < 0$ **then**
 $x_m(t+1) = x_m(t), s_m(t+1) = s_m(t) + s_l(t) + \Delta_l$
else
 $x_m(t+1) = x_m(t) + \min(\Delta_l, x_l(t) - x_m(t))$
 $s_m(t+1) = s_m(t) + s_l(t) + \max(0, \Delta_l - (x_l(t) - x_m(t)))$
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

3. $x_m(t) > x_l(t)$

(This last scenario is similar to the previous one).

Algorithm 15 Alternative Strategy III (Replacement 3.)

Data: $t, \mathcal{S}(t), \mathbf{r}(t), \mathbf{x}(t), \mathbf{s}(t)$
agent receiving the message m , leaving agent l
 $\Delta_l = x_l(t) - r_l$
if $\Delta_l \approx 0$ or $\Delta_l > 0$ **then**
 $x_m(t+1) = x_m(t), s_m(t+1) = s_m(t) + s_l(t) + \Delta_l$
else
 $x_m(t+1) = x_m(t) + \max(\Delta_l, x_l(t) - x_m(t)),$
 $s_m(t+1) = s_m(t) + s_l(t) + \min(0, \Delta_l - (x_l(t) - x_m(t)))$
end if
New agent $e \notin \mathcal{S}(s), \forall s \leq t (r_e = x_e(t))$
 $\mathcal{S}(t+1) = (\mathcal{S}(t) \cup \{e\}) \setminus \{l\}$

5.4.2 Analysis of the Process

Let us now consider the numerical results throughout different processes of this new strategy expected to solve the general problem. The results are represented on figures

5.6 and 5.7 corresponding respectively to a constant distribution and to strongly evolving distribution of the incoming values.

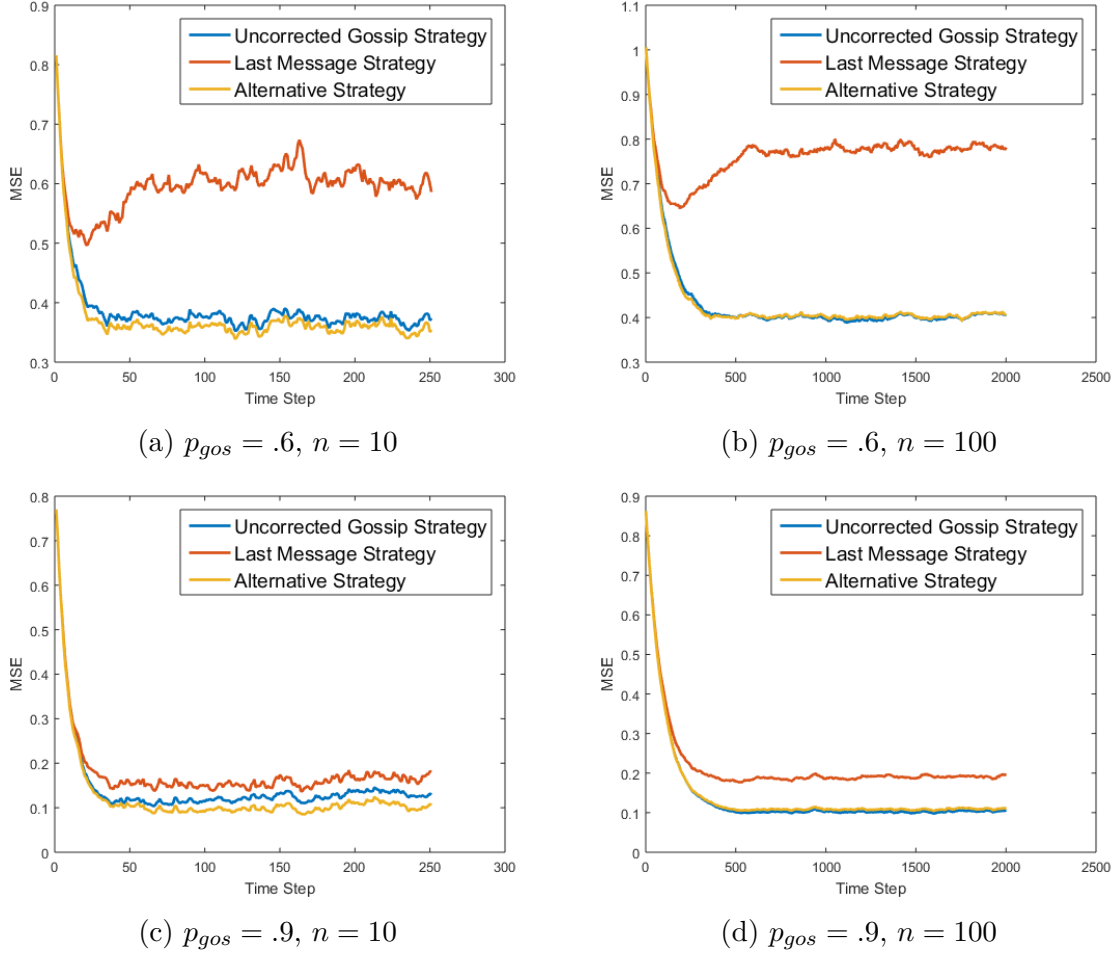
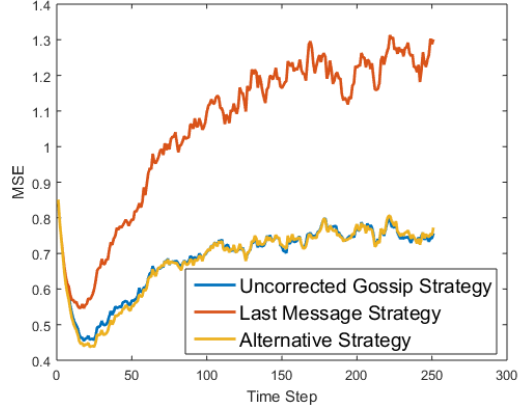


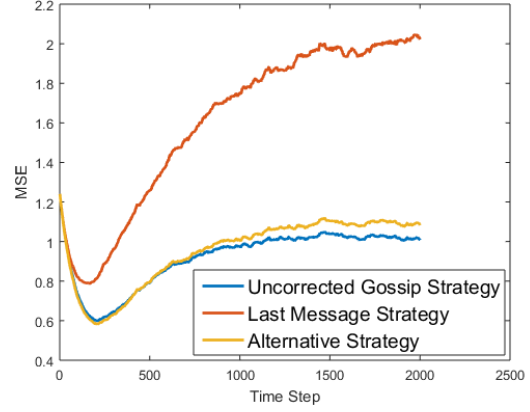
Figure 5.6: Results corresponding to the general problem (constant distribution).

This new strategy, improves the last message strategy as it seems to lead to smaller results, whatever the size of the system or the gossip rate. This confirms as expected that it is useful to avoid to change more than one approximation in the system as a replacement occurs. However, this new strategy is only an upgrade of the uncorrected gossip strategy in small systems. Otherwise, the errors of both strategies are similar in most situations. Indeed, in small systems, the slow inclusion of the reserves enables to decrease the group error between two replacements that explains the slightly better results of this new strategy. When the size of the system increases, the results of both methods get close to one another. At a replacement step, two events may occur: either the message is entirely stored as reserve and therefore the replacement step is similar to the uncorrected gossip strategy. If however, a portion of the last message is included in the reserve, the system tends to follow the same behaviour: the agent m takes the approximation of the leaving agent (or at least comes as close as possible), the approximation of agent m is therefore somehow replaced by the value of the incoming agent implying that both methods deal with the replacements in a similar way.

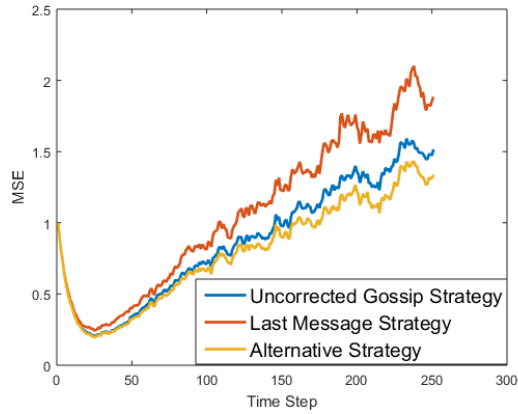
The main difference between the uncorrected gossip strategy and the new one is the use of the reserves that has an effect on the gossip steps. They are indeed a source of possible



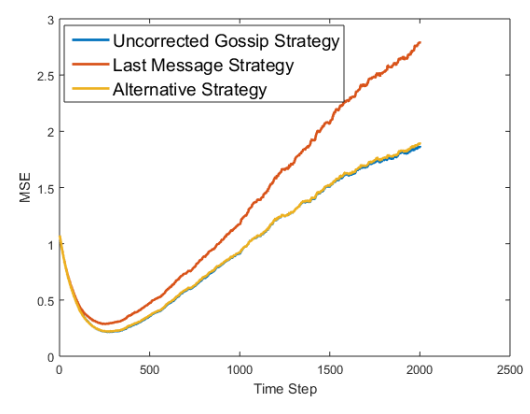
(a) $p_{gos} = .6, n = 10$



(b) $p_{gos} = .6, n = 100$



(c) $p_{gos} = .9, n = 10$



(d) $p_{gos} = .9, n = 100$

Figure 5.7: Results corresponding to the general problem in the design of an alternative strategy ($\mu(t)$ increases linearly up to 10).

raise of the disagreement as they get included in the system. Given the small value chosen for their inclusion, their effect is hardly felt.

When the distribution evolves through time, this new strategy seems however to loose efficiency. Indeed, this new strategy seemed to work slightly better than the uncorrected gossip strategy in small systems, or lead to similar errors when the system grows. When the distribution evolves, those behaviors remain only if the gossip rate is large. In that case, the reserves are included frequently in the system and decrease the group error.

5.4.3 Analysis of the Asymptotic Error

Let us finally present the asymptotic behavior of this new strategy function of the gossip rate. This is represented on figure 5.8. This graph is highly similar to the uncorrected gossip strategy and we will therefore not present the dependence to the size of the system.

This graph represents indeed the results observed in this last section. The decrease of the asymptotic error function of the gossip rate seems almost linear whatever the size of the system. On the contrary with the asymptotic results of the uncorrected gossip strategy

presented in chapter 3, the error tends to zero as the gossip rate tends to 1. This comes from the fact that the agents may decrease the group error when they gossip in that new strategy, which was not possible with the uncorrected gossip algorithm. This explains therefore the smaller MSE observed in previous subsection as well as the similarities between the methods when the system grows.

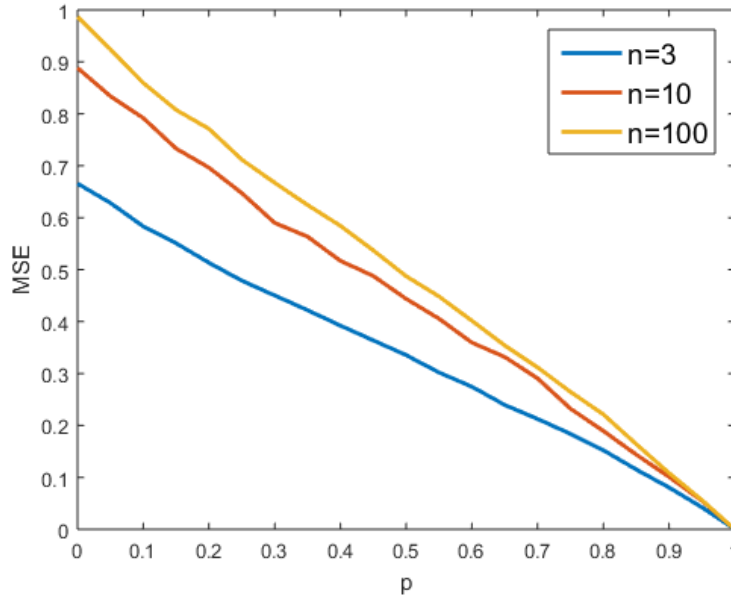


Figure 5.8: Asymptotic results of the new designed strategy function of the gossip rate.

5.5 Conclusion

The aim of this chapter was to design a new strategy that would improve the results of the last message strategy. The idea behind that alternative algorithm is to use the reserves and adapt the inclusion of the last message depending on the approximations of both agents interacting during a replacement, in order to get a disagreement as low as possible as the replacements occur. We have indeed seen that it was the main weakness of the last message strategy.

In this chapter, we first considered simplified scenario where the leaving agents interact with the incoming ones. The results showed hope to design a strategy adapted to the general problem that could lead to smaller errors than the uncorrected gossip strategy. We had indeed seen that the last message strategy could be improved using reserves in the previous chapter. Indeed, in that first case, the well chosen inclusion of the last messages enabled to keep a small disagreement due to the replacements, and the inclusion of the reserves conserved the mean of the system inside it.

There comes then the design of a general strategy based on the information done in the first cases. It has been seen that the last messages and reserves, if properly used, may be strongly decreasing the results of the last message algorithm. Those results came from the effort to change only one value in the system as the last messages were used. This improvement is so important that it leads to results that are similar to those of the

uncorrected gossip strategy in large systems, and that may even lead to lower errors in the case of small systems.

However, while the group error is kept low in small systems and large gossip rate (that was the weakness of the uncorrected gossip strategy) due to the (slow) inclusion of the reserves, it is however present in larger systems. The reserves are indeed not null in that case but their inclusion and storage function as the evolution of the group error in the uncorrected gossip strategy, by balancing themselves. It keeps nevertheless the advantage to lead to a zero group error for sure if the system closes long enough to include them back entirely in the system.

Conclusion

That master thesis had the first goal to analyze two initial algorithms being the uncorrected gossip and last message strategies. The former seemed to present flaws that the latter was expected to fulfill. Indeed, the fact that agents do not announce their departure in the former algorithm creates an error in the system that is assimilated to a group error. Moreover, if the system closes after a while, the error does not drop to zero using the uncorrected gossip algorithm. The last message strategy was therefore designed to avoid those unwanted phenomena as the agents send a well-chosen message to an agent in the system as they leave it.

Actually, while the theoretical expectations are in favour of the last message strategy, the numerical results are way different. Indeed, the broadcast of the last message that seemed to be an advantage, considering the group error, is in fact an important drawback regarding the disagreement in the system. It has been seen that the group error, that is nevertheless a problem in small systems, is quickly decreasing and becomes insignificant compared with the disagreement term when the systems grow. Therefore the perturbation introduced by the last messages is useless, especially since, in large systems or for small gossip rates, the agents are more rarely included in gossips. Therefore, the error linked with a message is likely to remain in the system for a long time, and therefore the associated error also does. There still remained hope that the last message strategy, could behave with more accuracy than the uncorrected gossip algorithm for processes where the mean of the distribution of the incoming values increases with time. Once again, we had to draw similar conclusions. The lack of announcement of the departure of the agents is not problematic compared with the disagreement introduced in the system by the messages even if the group error may become consequent in certain cases.

Given the bad results coming from the last message strategy, there rose the will to improve it. The idea was therefore to use reserves. This reserve-based strategy consists of storing the messages received in order to avoid the sudden raise of disagreement that can remain in the system for a long time. The surpluses (reserves) of each agent is then slowly included into the approximations of the agents during the gossip steps. Of course, this strategy is an improvement of the last message strategy since it is possible to choose the coefficient of that algorithm so that the results of this reserve-based strategy are similar to those of the uncorrected gossip algorithm (i.e. the reserves are never included in the system). However, this method presents its limits as well. If the coefficients associated to the reinsertion of the reserves are too small, the values stored in reserves are therefore source of a group error. If those coefficients grow, it is the disagreement that becomes problematic. It eventually comes that this strategy may be interesting in small systems but on the long run, the uncorrected gossip strategy provides the lowest errors.

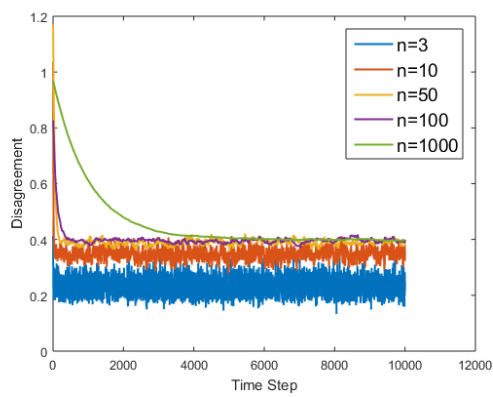
It therefore came to mind to design a strategy where the agents do not adopt a systematic behavior against the last messages. Instead of each time including or not the last messages to their own approximation in the same proportions, the agents to whom the messages are sent make themselves the decision how to deal with them. They make indeed a choice regarding the magnitude of the messages, their approximation as well as those of the agents broadcasting the messages. The goal is to decrease the disagreement resulting of a replacement step. This method is a real improvement of the last message strategy. The idea behind this new strategy was indeed to use adequately the last messages and the reserves in order to decrease the disagreement of a replacement step, as long as it is possible, by affecting only one approximation in the system if such an event occurs. This strategy presents therefore the good behaviour of the last message strategy in the small systems, and lead to results that are similar to the uncorrected gossip strategy in large ones, with the advantage to keep the mean of the system into it.

To conclude, the uncorrected gossip strategy presented potentially weaknesses that appeared to be valuable strengths. The different algorithms meant to fulfill those flaws are in general presenting unwanted characteristics, such as large disagreement or presence of an unwanted group error. They are nevertheless interesting to use in certain cases as in very small systems, or when the system is subject to a potential closure. However, when systems grow, those strategies quickly become inefficient.

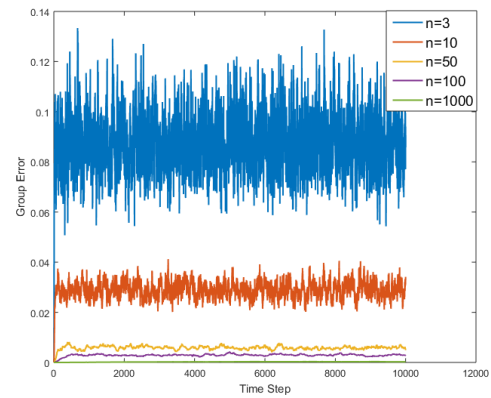
Appendix A

Additional Results

You can find in this appendix some results and graphs associated to chapters 3. They present however no additional or useful information to the flow of the main matter and are therefore only given here for curiosity.

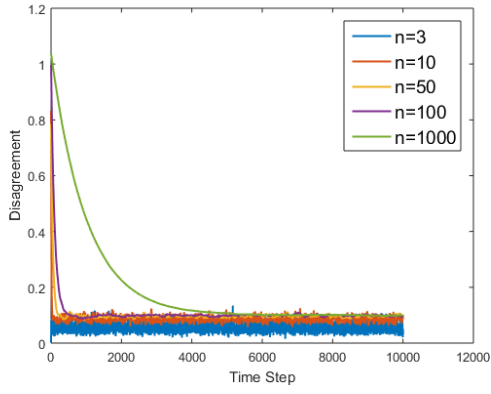


(a) Disagreement

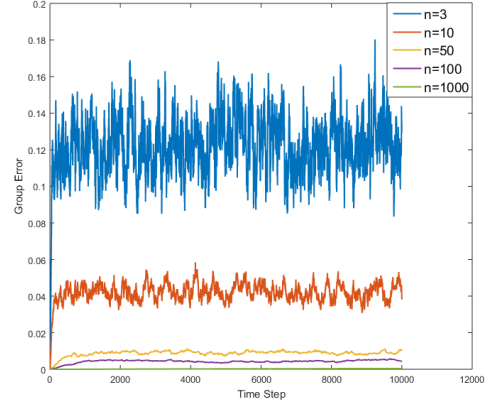


(b) Group Error

Figure A.1: Dependence on the number of agents for the uncorrected gossip strategy for a gossip rate of .6

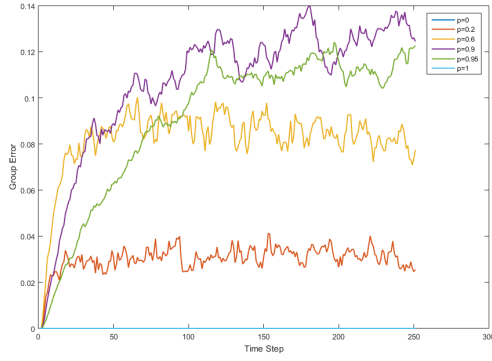


(a) Disagreement

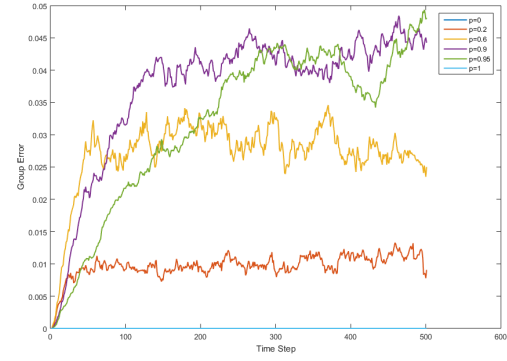


(b) Group Error

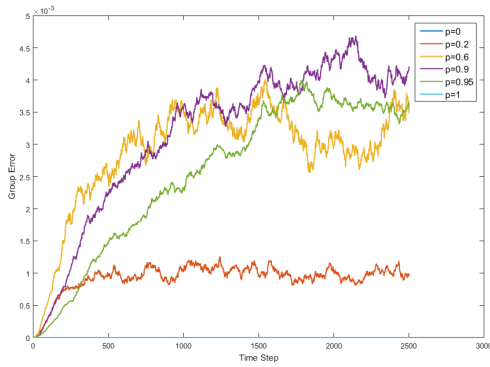
Figure A.2: Dependence on the number of agents for the uncorrected gossip strategy for a gossip rate of .9



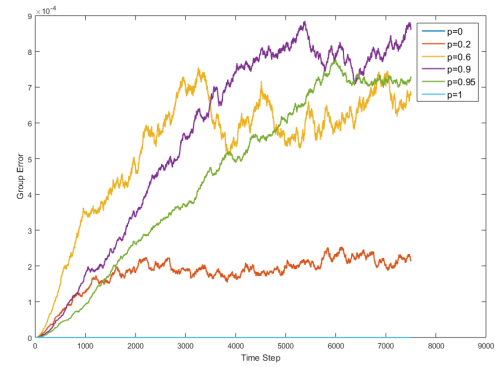
(a) $n = 3$



(b) $n = 10$

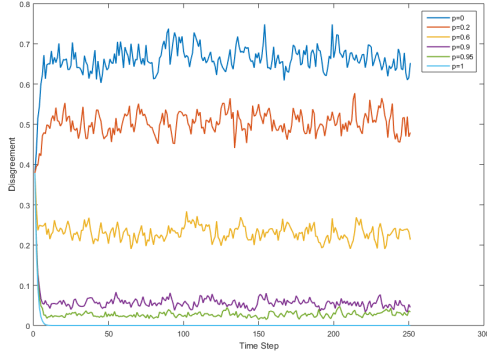


(c) $n = 100$



(d) $n = 500$

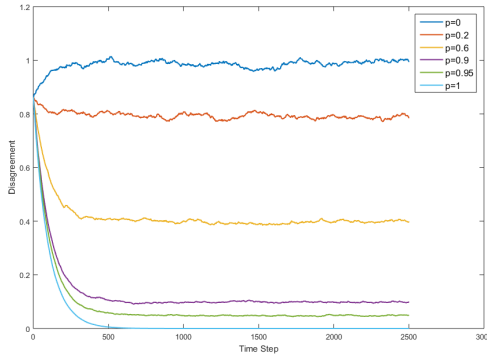
Figure A.3: Group error in the uncorrected gossip strategy for different systems



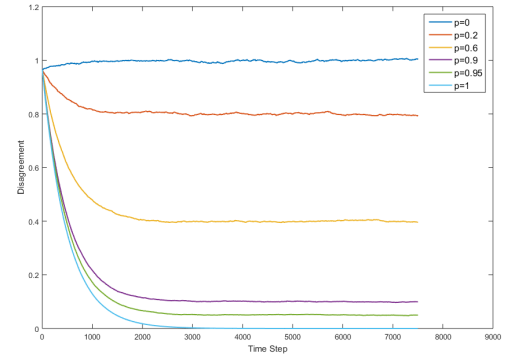
(a) $n = 3$



(b) $n = 10$



(c) $n = 100$



(d) $n = 500$

Figure A.4: Disagreement in the uncorrected gossip strategy for different system sizes

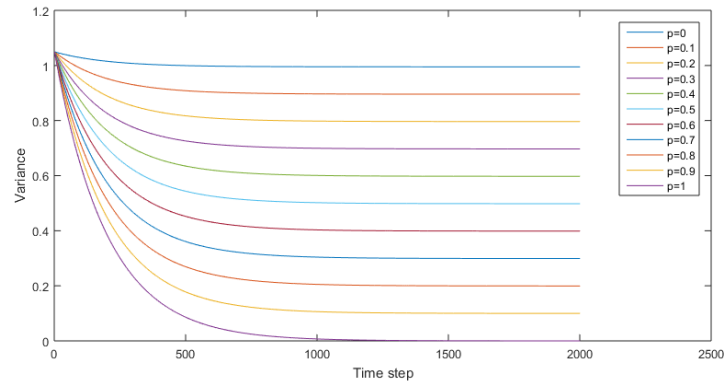


Figure A.5: Theoretical evolution of the variance throughout a process for 200 agents in the system and considering several gossip rates.

Appendix B

Notations and Symbols

n	number of agents in the system
$S(t)$	set of agents present in the system at time t
r_i	own value of agent i
$\bar{r}(t)$	mean of the own values of the agents present in the system at time t $(\frac{1}{n(t)} \sum_{i \in S(t)} r_i)$
$x_i(t)$	approximation of the mean of the system made by agent i at time t
$\bar{x}(t)$	mean of the approximations made by the agents present in the system at time t $(\frac{1}{n(t)} \sum_{i \in S(t)} x_i(t))$
$\Delta_l(t)$	variation of the approximation of an agent l in the system from its arrival until time t $(x_l(t) - r_l)$
$s_i(t)$	surplus (or reserve) stored by agent i at time t

Note also that indices i and j will commonly refer to agent included in a gossip step. During a replacement step, indices l , e refer respectively to the leaving and incoming agents. If the algorithm implies a last interaction between a random agent in the system and the replaced agent, index m refers to that one.

Appendix C

Matlab Codes

You can find in the appendix the three functions used in the analysis of the numerical results of the different strategies. The Chapters 3 and 4 are associated to only one function. Three functions are used in Chapter 5, one for each considered case.

Uncorrected Gossip Strategy and Last Message Strategy

BasicStrategies.m

```
1 %Function used to analyze the numerical results of the
2 %uncorrected gossip and last messages algorithms in chapter 3.
3 %
4 %INPUT: rInit->vector holding the initial values in the
5 %system, pgos->gossip rate, tmax->number of time steps of the
6 %process
7 %
8 %OUTPUT: MSE->array 2x(tmax+1) containing the MSE of the 2
9 %functions during a process:unc. gos. strat (line 1), last msg
10 %strat (line 2), Disagreement-> vector 1x(tmax+1) containing the
11 %disagreement in the unc. gos. strat (line 1)GroupError->
12 %vector 1x(tmax+1) containing the group error in the unc. gos.
13 %strat (line 1)
14 function [MSE,Disagreement,GroupError] =...
15     BasicStrategies(rInit,pgos,tmax)
16
17 n=length(rInit);
18 r=rInit;
19 xtUncorr=rInit;
20 xtLastMsg=rInit;
21
22 %Initialization of output vectors
```

```

23 MSE=zeros(2,tmax+1);
24 MSE(1,1)=mean((xtUncorr-mean(r)).^2);
25 MSE(2,1)=mean((xtLastMsg-mean(r)).^2);
26
27 Disagreement=zeros(1,tmax+1); GroupError=zeros(1,tmax+1);
28 Disagreement(1)=mean((xtUncorr-mean(xtUncorr)).^2);
29 GroupError(1)=mean((mean(xtUncorr)-mean(r)).^2);
30 for t=1:tmax
31     event=rand;
32     if event<pgos
33         i=ceil(rand*n); % 2 random (different) agents in the
            system
34         j=ceil(rand*(n-1));
35         if j>=i
36             j=j+1;
37         end
38         xtUncorr(i)=(xtUncorr(i)+xtUncorr(j))/2;
39         xtUncorr(j)=xtUncorr(i);
40         xtLastMsg(i)=(xtLastMsg(i)+xtLastMsg(j))/2;
41         xtLastMsg(j)=xtLastMsg(i);
42     else
43         i=ceil(rand*n); %replaced agent
44         j=ceil(rand*(n-1));
45         if j>=i
46             j=j+1;
47         end
48         xtLastMsg(j)=xtLastMsg(j)+xtLastMsg(i)-r(i);
49         r(i)=0*t/tmax+randn(1,1);
50         xtUncorr(i)=r(i);
51         xtLastMsg(i)=r(i);
52     end
53     %Assert: conservation of the mean for the last message
            strategy
54     assert(abs(mean(r)-mean(xtLastMsg))<1e-10,...
55         'The means are not conserved. ERROR 1')
56     %Update of the errors of the different errors
57     MSE(1,t+1)=mean((xtUncorr-mean(r)).^2);
58     MSE(2,t+1)=mean((xtLastMsg-mean(r)).^2);
59     Disagreement(t+1)=mean((xtUncorr-mean(xtUncorr)).^2);
60     GroupError(t+1)=(mean(xtUncorr)-mean(r)).^2;
61 end
62 end

```

Reserve-Based Strategy

Reserve.m

```
1 %Function used to analyze the numerical results of the
2 %reserve-based algorithm in chapter 4.
3 %
4 %INPUT: rInit->vector holding the initial values in the
5 %system, pgos->gossip rate, tmax->number of time steps of the
6 %process, k1,k2->coefficients of the algorithm
7 %
8 %OUTPUT: MSE->array 3x(tmax+1) containing the MSE of the 3
9 %functions during a process:unc. gos. strat (line 1), last msg
10 %strat (line 2), res-based strat (line 3), Disagreement->
11 %array 2x(tmax+1) containing the disagreement in the unc.
12 %gos. strat (line 1) and res-based strat (line 2), GroupError->
13 %array 2x(tmax+1) containing the group error in the unc. gos.
14 %strat (line 1) and res-based strat (line 2)
15
16 function [MSE,Disagreement,GroupError] =...
17         Reserve(rInit,pgos,tmax,k1,k2)
18
19 n=length(rInit); r=rInit;
20 %Initialization of the output vectors
21 xtUnc=rInit; xtLastMsg=rInit; xtReserve=rInit;
22 reserve=zeros(1,n);
23 MSE=zeros(3,tmax+1);
24 MSE(1,1)=mean((xtUnc-mean(r)).^2); MSE(2,1)=MSE(1,1);
25 MSE(3,1)=MSE(1,1);
26
27 GroupError=zeros(2,tmax+1);
28
29 Disagreement=zeros(2,tmax+1);
30 Disagreement(1,1)=mean((rInit-mean(r)).^2);
31 Disagreement(2,1)=Disagreement(1,1);
32 for t=1:tmax
33     event=rand; %Choice of event for time step t
34     i=ceil(rand*n);% 2 random (different) agents in the system
35     j=ceil(rand*(n-1));
36     if j>=i
37         j=j+1;
38     end
39     if event<pgos %gossip step
40         xtUnc(i)=(xtUnc(i)+xtUnc(j))/2; xtUnc(j)=xtUnc(i);
41         xtLastMsg(i)=(xtLastMsg(i)+xtLastMsg(j))/2;
42         xtLastMsg(j)=xtLastMsg(i);
43         reserve(i)=(reserve(i)+reserve(j))/2; reserve(j)=
            reserve(i);
```

```

44     xtReserve(i)=(xtReserve(i)+xtReserve(j))/2+k1*reserve(i)
45     ;
46     xtReserve(j)=xtReserve(i);
47     reserve(i)=(1-k1)*reserve(i); reserve(j)=reserve(i);
48     if abs(reserve(i))<=0.01
49         xtReserve(i)=xtReserve(i)+reserve(i);
50         xtReserve(j)=xtReserve(i);
51         reserve(i)=0; reserve(j)=0;
52     end
53 else
54     re=0*t/tmax+randn(1,1); %modify for evolving
55     distribution
56     rl=r(i);
57     xtUnc(i)=re; xtLastMsg(j)=xtLastMsg(j)+xtLastMsg(i)-rl;
58     xtLastMsg(i)=re;
59     delta=xtReserve(i)-rl;
60     xtReserve(j)=xtReserve(j)+k2*delta;
61     reserve(j)=reserve(j)+(1-k2)*delta+reserve(i);
62     reserve(i)=0;
63     xtReserve(i)=re;
64     r(i)=re;
65 end
66 %Assert: conservation of the mean for the last message and
67 %reserve-based strategies
68 assert(abs(mean(r)-mean(xtLastMsg))<1e-10,...
69     'The means are not conserved. ERROR 1')
70 assert(abs(mean(r)-mean(xtReserve+reserve))<1e-10,...
71     'The means are not conserved. ERROR 2')
72 %Update of the errors of the different errors
73 MSE(1,t+1)=mean((xtUnc-mean(r)).^2);
74 MSE(2,t+1)=mean((xtLastMsg-mean(r)).^2);
75 MSE(3,t+1)=mean((xtReserve-mean(r)).^2);
76 Disagreement(1,t+1)=mean((xtUnc-mean(xtUnc)).^2);
77 Disagreement(2,t+1)=mean((xtReserve-mean(xtReserve)).^2);
78 GroupError(1,t+1)=mean((mean(xtUnc)-mean(r)).^2);
79 GroupError(2,t+1)=mean((mean(xtReserve)-mean(r)).^2);
80 end
81 end

```

Alternative Strategy

Each of the 3 following functions corresponds to the three cases analyzed in chapter 5.
The general problem being dealt with function **Alternative3.m**

Alternative1.m

```
1 %Function used to analyze the numerical results of the second
2 %alternative algorithm in chapter 5.
3 %
4 %INPUT: rInit->vector holding the initial values in the
5 %system, pgos->gossip rate, tmax->number of time steps of the
6 %process
7 %
8 %OUTPUT: MSE->array 3x(tmax+1) containing the MSE of the 3
9 %functions during a process:unc. gos. strat (line 1), last msg
10 %strat (line 2), altern strat (line 3), Disagreement->
11 %array 2x(tmax+1) containing the disagreement in the unc.
12 %gos. strat (line 1) and altern strat (line 2), GroupError->
13 %array 2x(tmax+1) containing the group error in the unc. gos.
14 %strat (line 1) and altern strat (line 2)
15 function [MSE,GroupErr,Disagreement] =...
16 Alternative1(rInit,pgos,tmax)
17 n=length(rInit); r=rInit; k1=1;
18 xtUncorr=rInit; xtLastMsg=rInit; xtAltern=rInit;
19
20 %Initialization of the output vectors
21 MSE=zeros(3,tmax+1);
22 MSE(1,1)=mean((xtUncorr-mean(r)).^2); MSE(2,1)=MSE(1,1); MSE
23 (3,1)=MSE(1,1);
24 GroupErr=zeros(2,tmax+1); Disagreement=zeros(2,tmax+1);
25 Disagreement(1,1)=mean((rInit-mean(r)).^2);
26 Disagreement(2,1)=Disagreement(1,1);
27 s=zeros(1,n);
28 for t=1:tmax
29 event=rand;
30 i=ceil(rand*n); % 2 random (different) agents in the system
31 j=ceil(rand*(n-1));
32 if j>=i
33     j=j+1;
34 end
35 %Change following line according to n (to reach consensus)
36 if t~=10:10:90 % Gossip
37     s(i)=(s(i)+s(j))/2; s(j)=s(i);
38     xtUncorr(i)=(xtUncorr(i)+xtUncorr(j))/2;
39     xtUncorr(j)=xtUncorr(i);
40     xtLastMsg(i)=(xtLastMsg(i)+xtLastMsg(j))/2;
```

```

40     xtLastMsg(j)=xtLastMsg(i);
41     xtAltern(i)=(xtAltern(i)+xtAltern(j))/2+k1*s(i);
42     xtAltern(j)=xtAltern(i);
43     s(i)=(1-k1)*s(i); s(j)=s(i);
44     if abs(s(i))<=0.01
45         xtAltern(i)=xtAltern(i)+s(i); xtAltern(j)=xtAltern(i);
46         s(i)=0; s(j)=0;
47     end
48 else %replacement
49     rl=r(i); meanr=mean(r); Msg=xtAltern(i)-r(i)+s(i);
50     r(i)=10*t/tmax+randn(1,1); re=r(i);
51     xtUncorr(i)=r(i); xtLastMsg(i)=re+xtLastMsg(i)-rl;
52     if abs(rl-re)<0.1
53         xtAltern(i)=re+Msg; s(i)=0;
54     elseif rl>re
55         if abs(Msg)<0.05 || Msg>=0
56             xtAltern(i)=re+Msg; s(i)=0;
57         else
58             a=Msg/(re-rl); assert(a>0,'error: a<0!'); %>0
59             assert(k<=1,'error: k>1!');
60             if (a-1)/a>0
61                 xtAltern(i)=re+(a-1)/a*Msg; s(i)=(1-(a-1)/a)*Msg
62                 ;
63                 s(i)=(1-k)*Msg;
64             else
65                 xtAltern(i)=re; s(i)=Msg;
66             end
67         end
68     else
69         if abs(Msg)<0.05 || Msg<=0
70             xtAltern(i)=re+Msg; s(i)=0;
71         else
72             a=Msg/(re-rl); assert(a>0,'error: a<0!');
73             k=(1+(a-1)*n)/(a*n);
74             assert(k<=1,'error: k>1!');
75             if (a-1)/a>0;
76                 xtAltern(i)=(a-1)/a*Msg+re; s(i)=(1-(a-1)/a)*Msg
77                 ;
78                 s(i)=(1-k)*Msg;
79             else
80                 xtAltern(i)=re;
81                 s(i)=Msg;
82             end
83         end
84     end
85 %Assert: conservation of the mean for the last message and

```

```

86 %alternative strategies
87 assert(abs(mean(r)-mean(xtLastMsg))<1e-10,'Error Last Message
    Strategy');
88 assert(abs(mean(r)-mean(xtAltern)-mean(s))<1e-10,'Error
    Alternative Strategy');
89 %Update of the errors of the different errors
90 MSE(1,t+1)=mean((xtUncorr-mean(r)).^2);
91 MSE(2,t+1)=mean((xtLastMsg-mean(r)).^2);
92 MSE(3,t+1)=mean((xtAltern-mean(r)).^2);
93 Disagreement(1,t+1)=mean((xtUncorr-mean(xtUncorr)).^2);
94 Disagreement(2,t+1)=mean((xtAltern-mean(xtAltern)).^2);
95 GroupErr(1,t+1)=(mean(xtUncorr)-mean(r)).^2;
96 GroupErr(2,t+1)=(mean(xtAltern)-mean(r)).^2;
97 end
98 end

```

Alternative2.m

```

1 %Function used to analyze the numerical results of the second
2 %alternative algorithm in chapter 5.
3 %
4 %INPUT: rInit->vector holding the initial values in the
5 %system, pgos->gossip rate, tmax->number of time steps of the
6 %process
7 %
8 %OUTPUT: MSE->array 3x(tmax+1) containing the MSE of the 3
9 %functions during a process:unc. gos. strat (line 1), last msg
10 %strat (line 2), altern strat (line 3), Disagreement->
11 %array 2x(tmax+1) containing the disagreement in the unc.
12 %gos. strat (line 1) and altern strat (line 2), GroupError->
13 %array 2x(tmax+1) containing the group error in the unc. %gos.
    strat (line 1) and altern strat (line 2)
14 function [MSE,GroupErr,Disagreement] =...
15 Alternative2(rInit,pgos,tmax)
16 n=length(rInit); r=rInit; k1=0.2; k2=.1;
17 xtUncorr=rInit; xtLastMsg=rInit; xtAltern=rInit;
18 % Initialization of the output vectors
19 MSE=zeros(3,tmax+1);
20 MSE(1,1)=mean((xtUncorr-mean(r)).^2); MSE(2,1)=MSE(1,1);
21 MSE(3,1)=MSE(1,1);
22 GroupErr=zeros(2,tmax+1); Disagreement=zeros(2,tmax+1);
23 Disagreement(1,1)=mean((rInit-mean(r)).^2);
24 Disagreement(2,1)=Disagreement(1,1);
25 s=zeros(1,n);
26 for t=1:tmax
27 event=rand;
28 i=ceil(rand*n); % 2 random (different) agents in the system
29 j=ceil(rand*(n-1));

```

```

30 if j>=i
31     j=j+1;
32 end
33 if event<pgos %gossip
34     s(i)=(s(i)+s(j))/2; s(j)=s(i);
35     xtUncorr(i)=(xtUncorr(i)+xtUncorr(j))/2;
36     xtUncorr(j)=xtUncorr(i);
37     xtLastMsg(i)=(xtLastMsg(i)+xtLastMsg(j))/2;
38     xtLastMsg(j)=xtLastMsg(i);
39     xtAltern(i)=(xtAltern(i)+xtAltern(j))/2+k1*s(i);
40     xtAltern(j)=xtAltern(i);
41     s(i)=(1-k1)*s(i); s(j)=s(i);
42     if abs(s(i))<=0.01
43         xtAltern(i)=xtAltern(i)+s(i); xtAltern(j)=xtAltern(i);
44         s(i)=0; s(j)=0;
45     end
46 else %replacement
47     rl=r(i); delta_l=xtAltern(i)-rl;
48     %change following line for time-dep. Distribution
49     r(i)=0*t/tmax+randn(1,1); re=r(i);
50     xtUncorr(i)=r(i); xtLastMsg(i)=re+xtLastMsg(i)-rl;
51     if abs(rl-re)<0.1
52         xtAltern(i)=re+delta_l;
53     elseif rl>re
54         if abs(delta_l)<0.1
55             xtAltern(i)=re+delta_l;
56         elseif delta_l>0
57             xtAltern(i)=re+delta_l;
58         else
59             a=delta_l/(re-rl); assert(a>0,'error: a<0!');
60             assert(k<=1,'error: k>1!');
61             if (a-1)/a>0;
62                 xtAltern(i)=(a-1)/a*delta_l+re;
63                 s(i)=(1-(a-1)/a)*delta_l+s(i);
64             else
65                 xtAltern(i)=re; s(i)=delta_l+s(i);
66             end
67         end
68     else
69         if abs(delta_l)<0.1 || delta_l<0
70             xtAltern(i)=re+delta_l;
71         else
72             a=delta_l/(re-rl); assert(a>0,'error: a<0!');
73             k=(1+(a-1)*n)/(a*n);
74             assert(k<=1,'error: k>1!');
75             if (a-1)/a>0;
76                 xtAltern(i)=(a-1)/a*delta_l+re;
77                 s(i)=(1-(a-1)/a)*delta_l+s(i);

```

```

78         else
79             xtAltern(i)=re; s(i)=delta_l+s(i);
80         end
81     end
82 end
83     if abs(s(i))<0.1
84         xtAltern(i)=xtAltern(i)+s(i); s(i)=0;
85     end
86 end
87 %Assert: conservation of the mean for the last message and
88 %alternative strategies
89 assert(abs(mean(r)-mean(xtLastMsg))<1e-10,'Error Last Message
    Strategy');
90 assert(abs(mean(r)-mean(xtAltern)-mean(s))<1e-10,'Error
    Alternative Strategy');
91 %Update of the errors of the different errors
92 MSE(1,t+1)=mean((xtUncorr-mean(r)).^2);
93 MSE(2,t+1)=mean((xtLastMsg-mean(r)).^2);
94 MSE(3,t+1)=mean((xtAltern-mean(r)).^2);
95 Disagreement(1,t+1)=mean((xtUncorr-mean(xtUncorr)).^2);
96 Disagreement(2,t+1)=mean((xtAltern-mean(xtAltern)).^2);
97 GroupErr(1,t+1)=(mean(xtUncorr)-mean(r)).^2;
98 GroupErr(2,t+1)=(mean(xtAltern)-mean(r)).^2;
99 end
100 end

```

Alternative3.m

```

1  %Function used to analyze the numerical results of the third
2  %alternative algorithm in chapter 5.
3  %
4  %INPUT: rInit->vector holding the initial values in the
5  %system, pgos->gossip rate, tmax->number of time steps of the
6  %process
7  %
8  %OUTPUT: MSE->array 3x(tmax+1) containing the MSE of the 3
9  %functions during a process:unc. gos. strat (line 1), last msg
10 %strat (line 2), altern strat (line 3), Disagreement->
11 %array 2x(tmax+1) containing the disagreement in the unc.
12 %gos. strat (line 1) and altern strat (line 2), GroupError->
13 %array 2x(tmax+1) containing the group error in the unc. gos.
14 %strat (line 1) and altern strat (line 2)
15
16 function [MSE,Disagreement,GroupError] =...
17 Alternative3(rInit,pgos,tmax)
18 n=length(rInit); r=rInit; k1=.2;
19 xtUncorr=rInit; xtLastMsg=rInit; xtAltern=rInit;
20

```

```

21 %Initialization of the output vectors
22 MSE=zeros(3,tmax+1);
23 MSE(1,1)=mean((xtUncorr-mean(r)).^2); MSE(2,1)=MSE(1,1); MSE
    (3,1)=MSE(1,1);
24 GroupErr=zeros(2,tmax+1); Disagreement=zeros(2,tmax+1);
25 Disagreement(1,1)=mean((rInit-mean(r)).^2);
26 Disagreement(2,1)=Disagreement(1,1);
27
28 s=zeros(1,n); % used in the alternative algo
29 for t=1:tmax
30     event=rand;
31     i=ceil(rand*n);% 2 random (different) agents in the system
32     j=ceil(rand*(n-1));
33     if j>=i
34         j=j+1;
35     end
36
37     if event<pgos % Gossip
38         s(i)=(s(i)+s(j))/2; s(j)=s(i);
39         xtUncorr(i)=(xtUncorr(i)+xtUncorr(j))/2; xtUncorr(j)=
            xtUncorr(i);
40         xtLastMsg(i)=(xtLastMsg(i)+xtLastMsg(j))/2; xtLastMsg(j)=
            xtLastMsg(i);
41         xtAltern(i)=(xtAltern(i)+xtAltern(j))/2+k1*s(i);
42         xtAltern(j)=xtAltern(i);
43         s(i)=(1-k1)*s(i); s(j)=s(i);
44         if abs(s(i))<=0.1
45             xtAltern(i)=xtAltern(i)+s(i); xtAltern(j)=xtAltern(i);
46             s(i)=0; s(j)=0;
47         end
48     else % Replacement
49         rl=r(i); xm=xtAltern(j); xl=xtAltern(i); delta_l=xl-rl;
50         r(i)=randn(1,1);
51         re=r(i); X=xl; xtAltern(i)=re;
52         xtUncorr(i)=r(i); xtLastMsg(j)=xtLastMsg(j)+xtLastMsg(i)-rl;
53         xtLastMsg(i)=re;
54
55         if abs(rl-X)<0.1 || abs(delta_l)<.1
56             xtAltern(j)=xm+delta_l;
57             s(j)=s(j)+s(i); s(i)=0;
58         elseif xm<X
59             if delta_l<0
60                 s(j)=s(j)+s(i)+delta_l;
61             else
62                 xtAltern(j)=xm+min(delta_l,X-xm);
63                 s(j)=s(j)+s(i)+max(0,delta_l-(X-xm));
64             end
65         else

```

```

66         if delta_l>0
67             s(j)=s(j)+s(i)+delta_l;
68         else
69             xtAltern(j)=xm+max(delta_l,X-xm);
70             s(j)=s(j)+s(i)+min(0,delta_l-(X-xm));
71         end
72     end
73     s(i)=0;
74     if abs(s(j))<0.1
75         xtAltern(j)=xtAltern(j)+s(j); s(j)=0;
76     end
77 end
78 %Assert: conservation of the mean for the last message and
79 %alternative strategies
80 assert(abs(mean(r)-mean(xtLastMsg))<1e-10,'Error Last Message
    Strategy');
81 assert(abs(mean(r)-mean(xtAltern)-mean(s))<1e-10,'Error
    Alternative Strategy');
82 %Update of the errors of the different errors
83 MSE(1,t+1)=mean((xtUncorr-mean(r)).^2);
84 MSE(2,t+1)=mean((xtLastMsg-mean(r)).^2);
85 MSE(3,t+1)=mean((xtAltern-mean(r)).^2);
86 Disagreement(1,t+1)=mean((xtUncorr-mean(xtUncorr)).^2);
87 Disagreement(2,t+1)=mean((xtAltern-mean(xtAltern)).^2);
88 GroupErr(1,t+1)=(mean(xtUncorr)-mean(r)).^2;
89 GroupErr(2,t+1)=(mean(xtAltern)-mean(r)).^2;
90 end
91 end

```


Bibliography

- [A.02] Deshmukh A. *Knowledge Management, Organizational Intelligence and Learning, and Complexity, Distributed Artificial Intelligence*, volume 1. University of Massachusetts-Amherst, USA, 2002.
- [AB18] David Angeli and Pierre-Alexandre Bliman. *Stability and Leaderless Multi-Agent Systems. Extension of a Result by Moreau*. June 7, 2018.
- [CI12] Kai Cai and Hideaki Ishii. Average Consensus on General Connected Digraph. March 2012.
- [Gla06] Dr. Mevludin Glavic. *Agents and Multi-Agent Systems: A Short Introduction for Power Engineers*. Technical report, University of Liege, May, 2006.
- [HM17] Julien M. Hendrickx and Samuel Martin. *Open Multi-Agent Systems: Gossiping with Random Arrivals and Departures*. December 2017.
- [Ren04] *The multi-agent rendezvous problem, 42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 2, January 2004.
- [RKJ16] Michal Kopys Rafal Krolikowski and Wojciech Jedruch. *Self-Organization in Multi-Agent Systems Based on Examples of Modeling Economic Relationships between Agents*. July 29, 2016.
- [So08] Stephen So. *Why is the sample variance a biased estimator*. September 2008.
- [VDBT10] Julien M. Hendrickx Vincent D. Blondel and John N. Tsitsiklis. *Continuous-time average-preserving opinion dynamics with opinion-dependent communications. SIAM Journal on Control and Optimization (SICON)*, 48:5214–5240, 2010.
- [Wei99] Gerhard Weiss. *Multiagent Systems: a Modern Approach to Distributed Modern Approach to Artificial Intelligence*. Massachusetts Institute of Technology, 1999.
- [XL17] Jing Xie and Cheng-Ching Liu. *Multi-agent systems and their applications. Journal of International Council on Electrical Engineering*, 7, 2017.

