# Distributed Learning For Control

A Thesis
Presented to
The Academic Faculty
by

**Diana Carolina Mahecha Rojas**

In Partial Fulfillment
of the Requirements for the Degree
Master of Electronic Engineering

Department of Electrical and Electronic Engineering
Faculty of Engineering
Universidad de los Andes
July 2009

# Acknowledgements

This work would not have been possible without the support and guide of my advisor Nicanor Quijano. I would like to thank him for showing me the amazing world of research and for giving me the opportunity to work with the GIAP research group.
I would like to thank my parents for being the base of my live and for supporting me. I would also thank to Ronald Joven for being with me in all this process.
Finally, I would like to thank Fernando Lozano and Fernando de la Rosa for their inputs.

# Contents

# List of Figures

# List of Tables

# Abstract

Applying learning to distributed agent control solutions is one option to overcome some difficulties of these systems. Here, we study some bio-inspired algorithms in order to evaluate the learning influence in a swarm of agents. These agents work cooperatively and they try to eliminate some targets in a given environment. The aim of this work is to develop an individual based approach of imitation dynamics to improve the performance of the agents, and to achieve optimal distributions under different scenarios. Our algorithm is compared against other classical swarm intelligence techniques used to solve this type of problems.

**Keywords:** Multi-agent systems, Distributed learning, Evolutionary game theory, Imitation dynamics

# Introduction

There are a lot of problems that are more treatable from a distributed perspective, rather than a centralized one (e.g., large scale problems). In the past years, distributed solutions that show some type of intelligence have been subject of research. Some of the proposed solutions use multiple independent identities or agents that work together to gather and process information in order to achieve some objective [1]. In the development of these systems, learning has been introduced to improve the future behavior based in previous experiences [2].

The principal aim of this work is to evaluate some bio-inspired strategies of learning, and how they can improve the behavior of a group of agents. We consider a general problem in which we have a dynamical environment with targets (e.g., toxic substances), and a group of agents who works cooperatively and try to eliminate them. There are many proposed applications for this type of approach such as resource allocation [3], coordination to move obstacles [4], scheduling [5] [6] [7], exploring unknown environments and cooperative searching [8] [9], formation of particular shapes in ocean exploration [10], cooperative cleaning [11], and military surveillance and attack [12].

In order to control agents, there are some aspects that must be considered (e.g., communication between agents, or coordination and learning skills). In general, these aspects lead to complex configurations, mostly by the challenges of a reliable communication system [1][2]. However, in recent years, bio-inspired approaches have emerged as a

solution that simplifies these aspects. This kind of design seeks to exploit the evolved strategies of nature to construct high performance solutions [13]. Most of these techniques are inspired in social species (e,g., bees, ants, humans), and they offer an alternative to solve problems in a distributed way, without the provision of a global model of the system or complex communication systems. These techniques consider groups of homogeneous or heterogeneous individuals (agents) who seek to accomplish a common goal, which achieve emergent behaviors when they work cooperatively [13]. Besides, there is an interdependence of individual's benefits and costs, in which the efficiency of a particular strategy depends on both, individual and social behavior (social foraging) [14]. One important point is that the collective behavior is based only in elemental actions of each member's community (swarm intelligence) [15] [16], and by means of this, it is possible to achieve complex behaviors or tasks beyond individual's capacity. The more relevant characteristics to be exploited into an artificial agent swarm are: self-organization, indirect communication, adaptability, stability, flexibility, and robustness [15] [16].

When an agent or a group of agents start to learn, they start to play a more productive and active role [2]. In nature, learning is used for social animals to know how to do things and to know how to discriminate the useful information [14]. Swarm intelligence is a way to apply the last type of learning to a group of agents. In these strategies, the interactions between individuals are really important to achieve the goal. Through them, private and social information are combined and this gives to each agent more information that it can sense by its own. Here, we want that the agents learn about which is the best region to go in order to distribute themselves across the environment consequently. In order to achieve this behavior, we implement a strategy based on imitation dynamics. Imitation dynamics is an evolutionary game technique that models the information transference (learning) through cultural interaction (social information)

[17] [18]. We select this approach because it controls the way in which the agents are distributed across the environmental regions (habitats), and because it tries to maximize the individual profit for each agent (fitness). If we can achieve this kind of distribution with our agents, we can do a parallel attack and decrease the time and energy spent in the mission. Besides that, the strategies used by imitation dynamics are easily and individually applied to each agent. Some control strategies that uses approaches on evolutionary game theory, and that resolve an allocation problem as the one of this work, are given in [19], [20], [21],[22], [23] and [5].

Here, we want to develop a different approach based on a decentralized decision system that achieves profitable distributions (i.e., each agent makes its own decisions). In consequence, we develop an individual-based approach of imitation dynamics. The idea is to create an algorithm that allows the agents to achieve good distributions, so that they allocate themselves to the regions according to their profit. This approach is based only in local knowledge in order to avoid complex communication or sensing systems that requires full knowledge of the environment. In order to compare our approach, we select three classical swarm intelligence approaches: i) the distributed gradient search strategy [24] [25] [26]; ii) the ant colony optimization (ACO) algorithm for zone allocation [15]; and iii) the genetic algorithm [13]. The first one is selected because it exploits all the swarm and foraging characteristic mentioned. The second one solves a similar problem to the one that is proposed here, and also it tries to achieve profitable distributions. Finally, the last one is a search strategy very used in the literature and it is known by its fast convergence.

The remainder of this document is organized as follows: The imitation dynamics and its individual based approach are explained in Chapter 1, where some experiments are also shown. In Chapter 2 the other swarm intelligence algorithms are introduced and a comparison between the performance of imitation dynamics and the performance of

them is made. The discussion of the results is presented in Chapter 3. Finally, Chapter 3 contains some concluding remarks and directions for future work.

# Chapter 1

# Imitation Dynamics

Imitation dynamics is an evolutionary game-theoretic approach that describes the social evolution of the behavior in a population of interacting agents [17]. In this approach, cultural interactions are used for information transference, instead of inheritance. This kind of strategy is used by human societies and consists of spreading successful strategies through imitation [27]. The main concept is based on the human capacity of learning by observation and it gives the ability to expand its own knowledge based on the information shown by others. This process have three adaption levels [16] [23]: i) individual learning (the agent decides when it is advantageous to change strategy); ii) social learning (the agent learns from its neighborhoods' strategies); and iii) the knowledge is spread to the rest of the group by local interactions (culture).

Before we use imitation dynamics as a solution of the proposed general problem, we need to describe the habitat selection as a game. Let us assume that each habitat (region) is equivalent to a strategy. Therefore, we have a set of $N$ pure individual strategies, where strategy $j$ corresponds to the agent staying in region $j$, ($j = 1, ..., N$). We assume a constant and homogeneous population of $P$ agents, where the proportion of individuals with strategy $j$ (i.e., how many animals are in the $j^{th}$ region with respect to the

entire population $P$), corresponds to $x_j$. The $N$ dimensional strategy simplex used to denote the population distributions is given by $\Delta = \{x = (x_1, x_2, ..., x_N) | \sum_{k=1}^{N} x_k = 1$, and $x_j \geq 0$, for all $1 \leq j \leq N\}$. In addition, each strategy have an associated payoff function $f_j$ that depends on the population distribution and the targets abundance on each region. This payoff is modeled by the logistic function and it is given by,

$$f_j(x_j) = r_j \left(1 - \frac{x_j}{K_j}\right) \tag{1.1}$$

where $r_j = f_j(0)$ is the intrinsic growth rate of the $j^{th}$ region, and $K_j$ is the carrying capacity for which individual payoff is zero (i.e., $f_j(x_j = K_j) = 0$). The carrying capacity value is related to the maximum number of agents in a region.

In order to apply imitation dynamics to the population, we want that the agents learn the best strategy and how to follow it, so that the success is increased for all individuals. In the literature, continuum models, that describes the imitation dynamics for $x_i$, are developed (e.g., [17] and [18]). Here, we use an individual approach based on the rules of these models in order to allow that each agent takes its own decisions, instead that a central control, given by the continuum model, decides how many agents would go from one region to another.

## 1.1 Continuum Models

These models have two basic elements [17]: i) the time rate at which agents review their strategy; and ii) the choice probability for a reviewing agent. We use $q_j$ to denote the review rate of a $j$-strategist agent, and we take $p_j^i(x)$ as the probability that a reviewing $j$-strategist agent switches to another strategy $i$, where $p_j(x) = [p_j^1(x), p_j^2(x), ..., p_j^N(x)]$ is the probability distribution over the set of $N$ pure strategies. Taking these elements into account, the population distribution changes of imitation dynamics are given by,

$$\dot{x}_j = \sum_{i \neq j} x_i q_i p_i^j - \sum_{i \neq j} x_j q_j p_j^i \tag{1.2}$$

The first term is the proportion of agents that goes to region $j$, and the second term is the proportion of agents that are leaving that region. Usually, $q_j$ is equal to one and the agents interactions are only controlled by the probability $p_j^i(x)$. However, when we do that, we are wasting an important tool to control the energy expended by the agents. A successful agent does not always need to change its strategy, so it should not review it all the time. If we use a $q_j \neq 1$, we can allow less successful agents review their strategies more frequently than successful ones. This is viable if we let $q_j$ be a strictly decreasing function of $f_j$ [17]. Here, we consider two options for the reviewing rates that differ in how much they restrict the reviewing rates of the successful agents. These options are given by Equation (1.3). The first option is a linearly decreasing function of $f_j$, where $\beta > 0$ and $\frac{\alpha}{\beta} \geq f_j$ for all $j$. The second one is an exponentially decreasing function of $f_j$, where $0 < \beta < 1$ gives the spreading of the exponential. This is a stronger restriction for the successful agents.

$$q_j = \begin{cases} \alpha - \beta f_j & \text{Linearly decreasing rate} \\ \exp(-\beta f_j) & \text{Exponentially decreasing rate} \end{cases} \tag{1.3}$$

In [18] different models to implement the probability $p_j^i(x)$ are proposed. We select two of them that consider non-ideal individuals (proportional and myopic ones) because they represent more realistic scenarios. The first model considers individuals with preferences proportional to differences in payoffs, and it is an optimal learning strategy when there is no more information available [28]. In this model, a probability distribution according to the difference between the payoffs of the region $j$ and a set of comparing regions is created. These regions are within a set, which is denoted by

$V_j$. This set depends on the amount of information that it is taken into account and we describe its selection later. For the proportional agents, the probability to change from strategy $j$ to strategy $i$ is given by

$$p_i^j(x_i) = \begin{cases} \mu(f_i - f_j) & \text{if } f_i > f_j, i \neq j, i \in V_j \\ 0 & \text{if } f_i < f_j, i \neq j, \text{ or } i \notin V_j \\ 1 - \sum_{f_k < f_j} \mu(f_k - f_j) & \text{if } i = j, \text{ and } k \in V_j \end{cases} \tag{1.4}$$

where $\mu = \frac{1}{\sum_{i \in V_j} |f_i - f_j|}$ is used to secure that $p_j$ is a probability distribution. The second model considers myopic individuals. This is a simple model because it uses individuals with local knowledge of their environment. We assume that each $j-$strategist agent randomly samples another habitat $i$, and switches to it when it offers a higher payoff. The sampled region $i$ should belong to the set of comparing strategies $V_j$ according to the interactions used (restricted or free). This model is given by,

$$p_j^i = \begin{cases} \frac{1}{|V_j|} & \text{if } f_i > f_j, i \neq j, \text{ and } i \in V_j \\ 0 & \text{if } f_i < f_j \text{ , } i \neq j, \text{ or } i \notin V_j \end{cases} \tag{1.5}$$

where $|V_j|$ is the number of elements of $V_j$. As we already said, the selection of $V_j$ depends on the amount of information that is taken in account. We consider two cases: (a) $V_j$ is composed by the neighboring regions (restricted interaction case); and (b) $V_j$ is composed by the rest of $(N-1)$ regions (free interaction case). Figure 1.1 shows the two cases for the both types of individuals in a grid of $3 \times 3$ regions. We can see that proportional individuals always use more information of the other regions that the myopic ones, although that the set $V_j$ is the same. In the free interaction case, the proportional individuals have to sense all the $8$ regions in $V_j$ (Figure 1.1(a)). On the other hand the myopic ones only have to sense one region that it aleatory select from

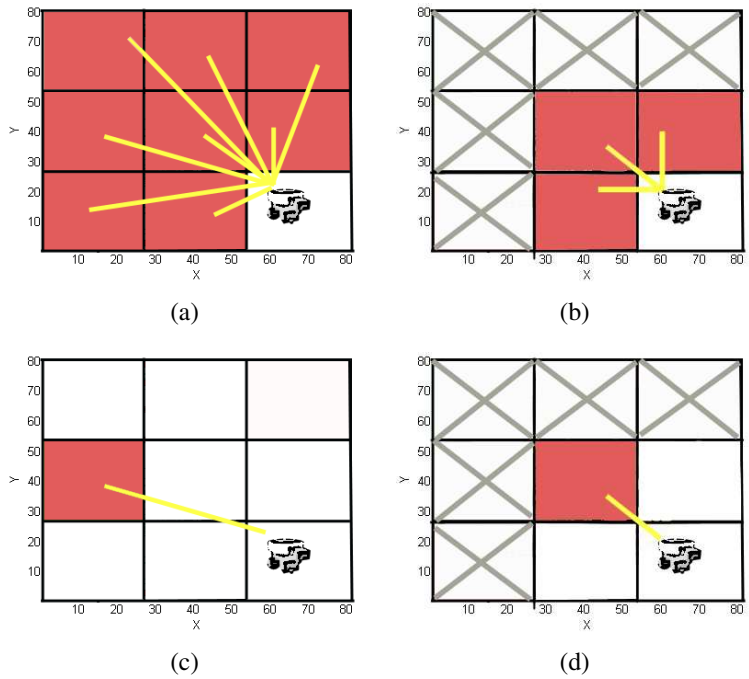



Figure 1.1: Information used by the proportional and myopic individuals for the two cases of $V_j$ (restricted and free interaction). Parts (a) and (b) show a proportional individual with free and restricted interactions, respectively. Parts (c) and (d) show a myopic individual with free and restricted interactions, respectively. The red regions are the regions sensed by each agent. The regions that have an X are not part of the set $V_j$.

$V_j$ (Figure 1.1(c)). In the restricted interaction case, the proportional individuals have to sense all the neighboring regions (Figure1.1(b)), whereas that the myopic ones only sense the region that they aleatory select (Figure 1.1(d)).

## 1.2 Individual Based Approaches

Imitation Dynamics gives a strategy that is easy to implement in a group of agents in order to they distribute themselves in the environment according to quality of each region. To do that in a distributed way, each agent should be capable of sensing its own fitness and the fitness of the other strategies in some way. The decision made by

each individual depends directly on this information, and the key of our approach is that this decision is autonomously and individually taken. We consider a swarm of $P$ homogeneous agents, and a point mass dynamics for them. The changes in the position of the agents only depend on where they want to go. This model has been used in [24] [26] and although it is very simple, it gives a good initial approach to the control of the agents. Assuming a constant velocity, we obtain a model given by,

$$\dot{p}_h = F_h, \quad h = 1, ..., P \tag{1.6}$$

where $F_h$ is the force or control input for the $h^{th}$ agent. This control input depends on the decisions that the agent takes. If an agent decides to go from a region $j$ to a random point on the $i^{th}$ region, the value of $F_h$ is given by the difference of both points. Otherwise, if the agent decides to stay in its region, it evaluates the point in which it is. If the targets in that point are depleted, it selects a random point into its own region, and the value of $F_h$ is the difference between the future and the actual point. We use this simple model to focus on the collective behavior emerged when the individual decision system based on imitation dynamics is applied (swarming convergence), but not to focus on the low level control of the physic and actuators of the agent. Although this is a strong assumption, we consider that our proposal should be tested first in this simple scenario because normally, some of the techniques used do not consider the physical characteristics of the agent.

In order to reproduce the type of strategy proposed by the imitation dynamics, we use the probabilities $q_j$ and $p_i^j$ defined before to implement the decision system of each agent. An agent that is in the $j^{th}$ region will review its strategy with a probability given by $q_j$ in Equation (1.3). If the agent reviews its strategy, it will go from region $j$ to

region $i$ with probability $p_j^i$. If we consider proportional individuals, each agent should create a probability distribution that evaluates the regions that belong to $V_j$ and that includes its own region. This probability distribution is given by Equation (1.4) and according to it, the agent decides to which region it should move. In the other hand, if we consider myopic individuals, the decision process is simpler. Each agent should randomly select a region $i \in V_j$ and it moves to it, only if the payoff associated to that region $f_i$ is grater that its own payoff $f_j$. This behavior is described in Equation (1.5). The algorithm implemented to simulate this individual based approach of the imitation dynamics is given in Algorithm 1.

**Algorithm 1** Individual based approach of Imitation Dynamics

```
Generate random initial positions for the P agents.
Initialize σ(p_h)
repeat
  for h = 1 to P do
    Remove γ in position p_h
    Calculate r_j, f_j and q_j where j is the strategy for the h^th agent
    if rand < q_j then
      Select a region i according to the strategy (Myopic or Proportional) to move
      in.
      if i ≠ j then
        Move to i.
      end if
    else
      if σ(p_h) > 0 then
        ṗ_h = F_h = 0
      else
        Select randomly a point (p_new) into region j
        Move to p_new. F_h = (p_new − p_h)
      end if
    end if
  end for
until t > Simulation time
```

| Symbol | Definition |
|---|---|
| $\sigma$ | Standard deviation of the data |
| $(\bar{f}_i^*)$ | Mean of the average fitness of the inhabited regions |
| $\sigma(\bar{f}_i^*)$ | Standard deviation between the average fitness of inhabited regions (it estimates the quality of the distribution) |
| $HR$ | Mean of the number of inhabited regions at equilibrium |
| $It$ | Convergence iteration |
| $IQR$ | Interquartile range of the $\bar{f}_i^*$ after the population achieves the equilibrium point. This last parameter is an estimate of the variations that occur after convergence. |
| $FL$ | Mean of the total final level of the targets. |

Table 1.1: Statistical parameters used to evaluate the distributions achieved

## 1.3 Simulation Results

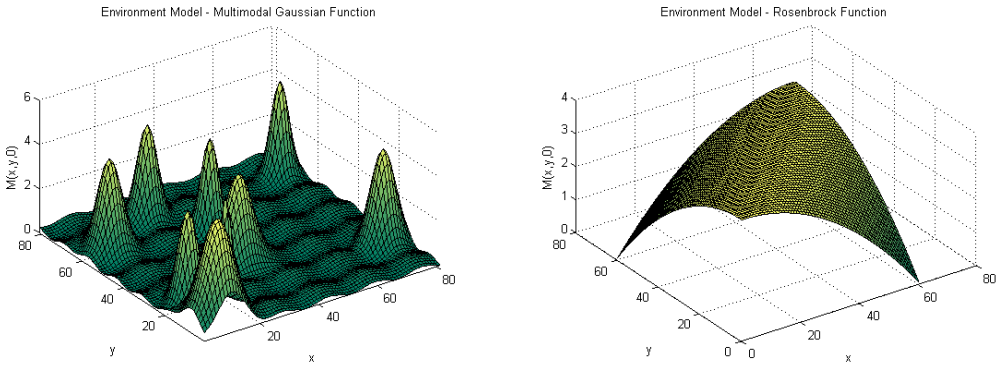
In order to illustrate the main characteristics of our algorithm, three experiments are performed. The first experiment evaluates the difference between the results of the continuum model and the individual based approach. The second one explores the influence of adding the interaction restrictions and the changes on $q_i$. It also evaluates the grid influence on the execution times of the algorithm. Finally, in the third experiment, the travel time between regions is taken into account. A population of $100$ agents is taken and simple grids, in which the interaction restriction have an effect, are used. All the experiments are realized through Monte Carlo simulations with $100$ runs. The results are evaluated with the average values of statistical parameters in Table 1.1. The behavior of this dynamics is evaluated in a model for the environment in which the agents move in, which is described next.

### 1.3.1 Environment Model

We assume that agents move over a resource profile $\sigma(\cdot)$ in a bidimensional space. This profile models the location and amount of targets in the environment (e.g., the toxic

substances). This kind of attractant/repellent profile is composed by artificial potential functions and it has been used in some works before [26] [25]. Besides target information, the model might include other factors of the environment, such as dangerous zones for the agents, or obstacles. However, in this case, we consider only the targets information. In the experiments we use two functions to prove different environment profiles. The first one is a multimodal Gaussian function, and the second one is the Rosenbrock function. The multimodal Gaussian profile is a combination of several Gaussian and small magnitude sinusoidal/cosinusoidal functions. The Gaussian functions are used to represent picks of different magnitudes for the targets that are centered in different points on the space. The other functions represent small magnitude maxima and minima. The multimodal Gaussian profile used is shown in Figure 1.2(a). On the other hand, the Rosenbrock function [29] represents an environment in which the maximum target levels are centered in the diagonal of the environment. This is a function commonly used to test optimization algorithms and it is shown in Figure 1.2(b). The multimodal Gaussian profile is used in the experiments shown in this chapter because it represents a more challenging scenario. The Rosenbrock function is more uniform and find the most profitable regions in an environment defined by it, it is not so difficult. We evaluate this function on the next Chapter.

Finally, we define a grid of $N$ regions in this environment. Each region is denoted by $j$ ($j = 1, 2, ...N$), and represents a possible strategy for the agents. The amount of resources in region $j$ is given by $r_j = f_j(0)$ (i.e., the available resources when any agent is in it). In the experiments $r_j$ corresponds to the sum of the values of the profile $\sigma()$ in the points that are into the region $j$. In the restricted interaction case, an agent that is in a region $j$ can only interact with the neighboring regions. For example, in a grid of $6 \times 6$, as the one showed in Figure 1.3.1, a $15_{th}$-strategy agent (i.e., an agent who is in region $15$), only can interact with the set of regions $V_{15} = 8, 9, 10, 14, 16, 20, 21, 22$.




(a) (b)

Figure 1.2: Environment models. The multimodal Gaussian profile is in part (a). Part (b) shows the Rosenbrock Function.

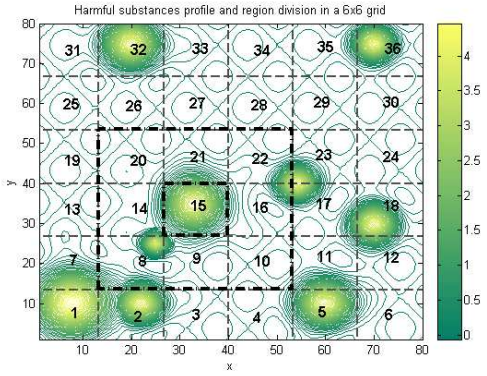
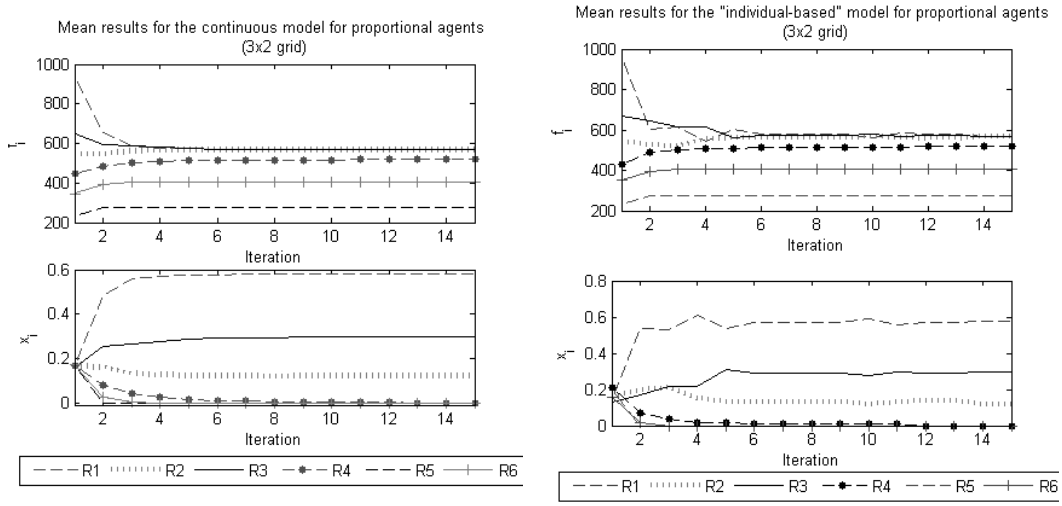



Figure 1.3: Multimodal Gaussian profile with a $6 \times 6$ grid and the interaction region for a $15$-strategist agent. Lighter lines represent the highest targets levels.

and $|V_{15}| = 8$.

### 1.3.2 Experiment No. 1 - Continuum vs. Individual Approach

In this experiment we run the continuum model and the agent based approach for the two types of individuals considered: myopic and proportional. We use the free interaction approach and $q_j = 1$ in order to can evaluate its influence on next experiments. We take a grid of $3 \times 2$ and the multimodal Gaussian profile in Figure 1.2(a). It is important to emphasize that this grid is selected only for convenience in the presen-

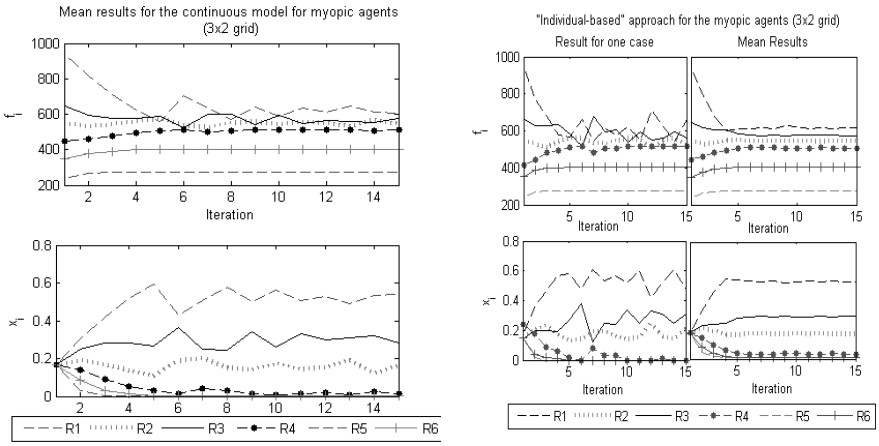



(a) (b)

Figure 1.4: Results for proportional individuals. Distributions and fitness functions achieved by: (a) continuum model, and (b) "individual-based" model.

tation of results, and that other dimension grids can also be used. With this configuration, the total amount of resources in each region is given by $r = [r_1, r_2, ..., r_6] = [1089, 630.97, 751.49, 518.26, 274.37, 403.41]$. The evolution of $f_j$ and $x_j$ for each region are shown in Figure 1.4 for the proportional case. We observe that in the continuum model the individuals are allocated between the most profitable regions (i.e., regions 1, 2, and 3). The proportion of individuals in each region for this case is given by $x = (0.5808, 0.2969, 0.1216, 0, 0, 0)$. The agent approach has a similar behavior and it achieves an equilibrium distribution with the same inhabited regions, i.e., $x = (0.58, 0.3, 0.12, 0, 0, 0)$.

For the case of myopic individuals, the same test is performed. The results are shown in Figure 1.5. We can see that the agent approach follows the behavior of the continuum model and that both models oscillate around the same equilibrium point. In this case a continuous equilibrium point is not achieved because the agents make their decisions at the same time. When this happens, all of them have the same information, and there is a high probability that a large proportion of them makes the same decision and moves




(a) (b)

Figure 1.5: Results for the myopic case. Distributions and fitness functions achieved by: (a) continuous models, and (b) "individual-based" models.

to the most profitable region. However, when they arrive to that region, their payoff decreases and the other regions, including the original one, could be more profitable. If this happens, a considerable portion of the same agents goes to other regions at the next iteration, and this cyclic behavior remains through all the process. As a result, after the equilibrium is achieved, the agents move continuously between regions with an oscillatory behavior around the equilibrium point. This behavior is presented in both models (for the proportional case appears when the complexity of the grid is increased), and it represents a problem because the agents are not able to meet any region's demand.

### 1.3.3 Experiment No. 2 - Restriction interaction and $q_i$ effects

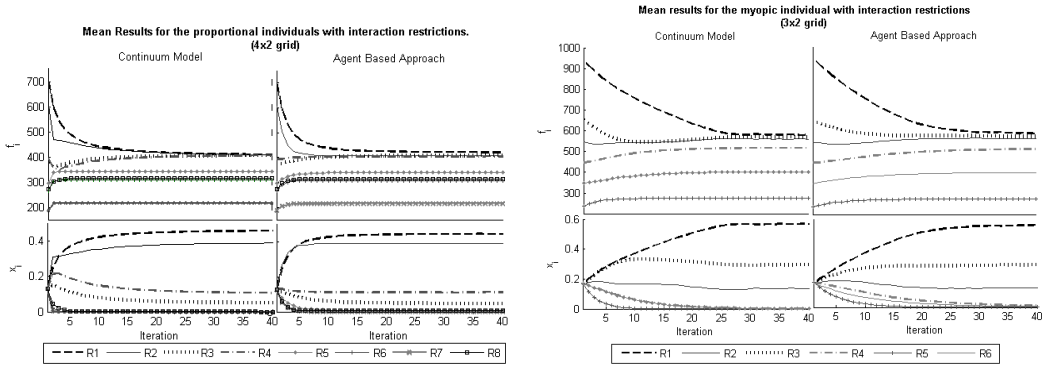
| Agents | $q_i$ | $\bar{\sigma}$ | $\overline{(\bar{f}_i^*)}$ | $\overline{\sigma(\bar{f}_i^*)}$ | $HR$ | $\overline{IQR}$ |
|---|---|---|---|---|---|---|
| Prop. Model | 1 | 0.0009 | 406.96 | 0.2096 | 4 | 2.5439 |
| Prop. Indiv. | 0.6 | 0.0158 | 409.96 | 7.2636 | 4 | 1.4967 |
| Myop. Model | 0.4 | 0.0088 | 569.48 | 9.3659 | 3 | 10.569 |
| Myop. Indiv. | 0.2 | 0.0165 | 571.7386 | 12.1058 | 3 | 14.8576 |

Table 1.2: Statistic Results for the Experiment No. 2.

In order to avoid the oscillatory behavior shown previously, we can change the value

of $q_j$. For this test, we consider the restricted interaction case, i.e., we do not have full information. As before, we evaluate the behavior presented for both type of individuals: myopic and proportional. Also the continuum model and individual based model are considered. We select the simplest grids that present some oscillator behavior with $q_j = 1$, and in which the interaction restrictions have an effect. For the myopic case is used a grid of $3 \times 2$ and for the proportional one is used a grid of $4 \times 2$. In general, when the value of $q_j$ decreases, the amount of oscillations at equilibrium point also decreases, but the convergence time increases. $q_j$ was changed between $0$ and $1$ and the parameters that present the best results are shown in Table 1.2. We can see that there are some differences between the behavior predicted by the model, and the behavior shown by the agents in both cases (myopic and proportional). In one hand, both models predict a final distribution in which all the inhabited regions have almost the same payoff (i.e., the $\overline{\sigma(\bar{f}_j^*)}$ value is small). In the other hand, we have that both agent based approaches achieve a distribution close to the one predicted by the models, but the final values of $f_j$ for the inhabited regions present some differences between them (i.e., the values of $\overline{\sigma(\bar{f}_j^*)}$ are bigger). This happens because the individual approach guarantees an integer number of agents in each region, whereas the continuum model does not. In the continuum model, the proportion of individuals that should be in each region is given, but it does not consider that a real agent can only be in one region. In some cases the proportions given arise in non-integral distributions (e.g., 45.5 agents are assigned to region $j$). Figure 1.6 shows the evolution of $f_j$ and $x_j$ for each case. We can see that the proportional approach converges faster that the proportional one and that the oscillations are controlled by $q_j$ in both cases. This last point is also shown by the values of $\overline{IQR}$, which are low if we compare them with the magnitudes of the inhabited regions $f_j$.

Finally, in Table 1.3 are showed the computational times for some grids and both type




(a) (b)

Figure 1.6: Results for the experiment No. 2. The mean values for the fitness and the distributions achieved are shown. Part (a) shows the proportional case. Part (b) shows the myopic case.

of individuals. The results for the proportional agents are in Parts 1.2(a) and 1.2(b). We can see that the computational time increase considerably whit the grid complexity. If we compare de free interaction case with the restricted results, we can see that restrict the interactions reduces the computational time for the agents, specially in more complex grids. This happens because the amount of information taken in account in each iteration is less when the restrictions are added. For example, in a $6 \times 6$ grid with free interactions, each agent should create a probability distribution over $36$ regions. However, when we add the restrictions, each agent create a probability distribution over maximum $8$ regions. The results for the myopic agents are in Parts 1.2(c) and 1.2(d). We can see that in general all the times are in the same range. This shows that the complexity of the grid do not have influence in the computational time of the myopic individuals because independent of the interactions used, the myopic agents only evaluate a region each time.

(a) Proportional - Free interaction

| $Grid$ | $Time$ | $\sigma(Time)$ |
|---|---|---|
| $2 \times 2$ | 4.7038 | 0.2326 |
| $3 \times 3$ | 8.3117 | 0.2594 |
| $4 \times 4$ | 13.8716 | 0.4522 |
| $5 \times 5$ | 24.3179 | 4.2480 |
| $6 \times 6$ | 35.1432 | 1.8591 |

(b) Prop. - Interaction Restriction

| $Grid$ | $Time$ | $\sigma(Time)$ |
|---|---|---|
| $2 \times 2$ | 3.8605 | 0.3797 |
| $3 \times 3$ | 4.2770 | 0.2932 |
| $4 \times 4$ | 4.9443 | 0.1872 |
| $5 \times 5$ | 5.3697 | 0.1872 |
| $6 \times 6$ | 6.5136 | 0.1761 |

(c) Myopic - Free Interaction

| $Grid$ | $Time$ | $\sigma(Time)$ |
|---|---|---|
| $2 \times 2$ | 2.8608 | 0.1572 |
| $3 \times 3$ | 2.7970 | 0.1155 |
| $4 \times 4$ | 2.8520 | 0.0906 |
| $5 \times 5$ | 2.8595 | 0.1073 |
| $6 \times 6$ | 2.9721 | 0.0906 |

(d) Myopic - Interaction Restriction

| $Grid$ | $Time$ | $\sigma(Time)$ |
|---|---|---|
| $2 \times 2$ | 2.9151 | 1.6161 |
| $3 \times 3$ | 2.6184 | 0.9296 |
| $4 \times 4$ | 2.4487 | 0.0545 |
| $5 \times 5$ | 2.5798 | 0.0762 |
| $6 \times 6$ | 2.7258 | 0.1681 |

Table 1.3: Results for the experiment No

### 1.3.4 Experiment No. 3 - Travel time influence

Until now, we have assumed that travel between regions has no cost for the agents. This is a very strong assumption, specially when we want to implement these strategies in a group of real agents. In this experiment we relax that assumption, and we consider a travel time between regions proportional to the distance between them. We also add a Gaussian disturbance at iteration $1000$ in order to evaluate the reaction to changes of the environment. We use a $3 \times 3$ grid in order to can evaluate both approaches under the same criteria and because a symmetrical grid gives a better division of a square environment (as the one we use). The disturbance is centered at point $(60, 50)$, and it has a spread of $0.06$, so it mainly changes the fitness values of regions $5$ and $6$. When there is a travel time, the oscillations presented before become an important problem and a strong restriction in $q_j$ is needed for successful agents. We test the linear and the exponential function of $q_j$ given in Equation (1.3), and we obtain that the exponential one gives a better result. Figure 1.7 shows some states of the evolution of this process. In Part (a) we see the initial position of the agents. Part (b) shows the

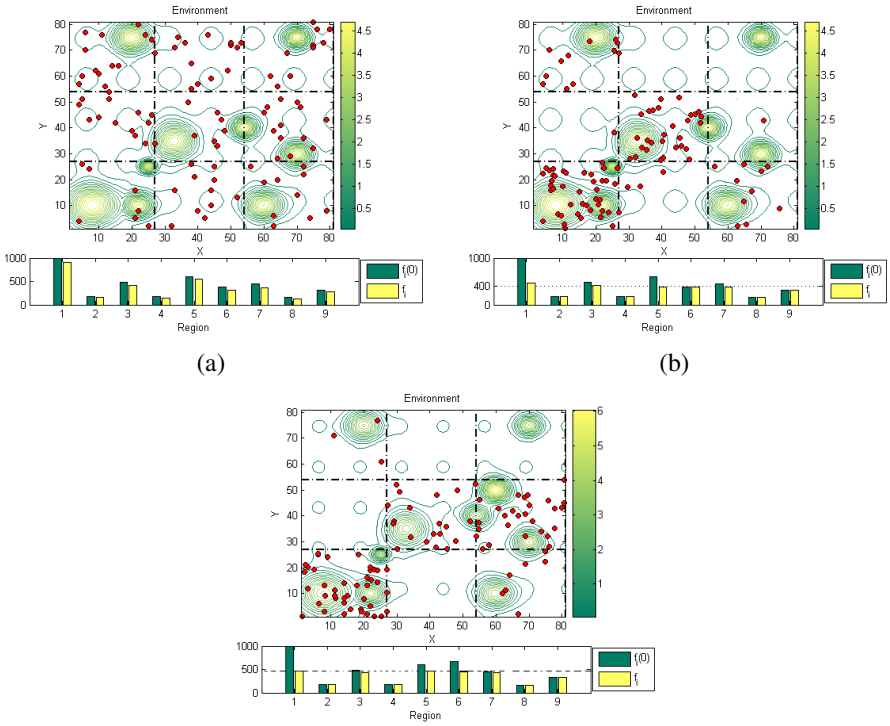
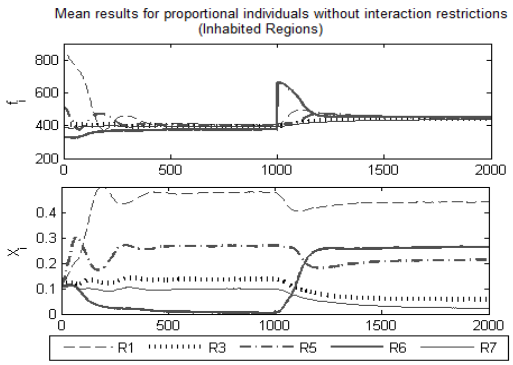


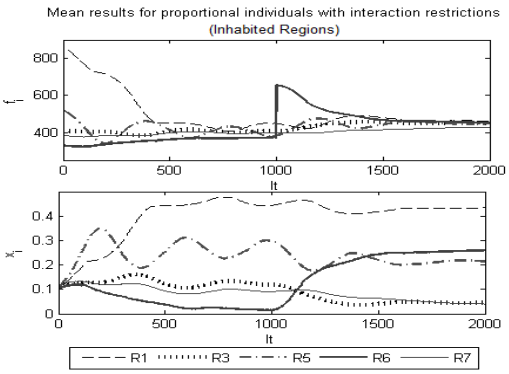

Figure 1.7:

distribution achieved by the agents before the perturbation and finally, Part (c) shows the perturbation added (regions 5 and 6) and how the agents distribute themselves around the new environment.

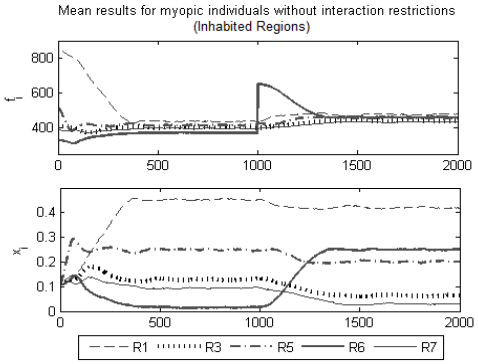
The evolution of $f_j$ and $x_i$ for the individual approach are shown in Figure 1.8. We can see that the value of the convergence iteration considerably increases for all cases, because the time travel makes the process slower. If we compare the two cases without interaction restrictions (Parts 1.8(a) and 1.8(c)) we can see that the proportional achieves an equilibrium distribution faster than the myopic case, before and after the perturbation. However, its computational cost almost doubles the cost of the myopic case. Thus, the proportional model gives a better response but it is not necessarily the best option because it requires a lot of calculations. When the interaction restrictions are included (Parts 1.8(b) and 1.8(d)), the computational cost of the proportional case



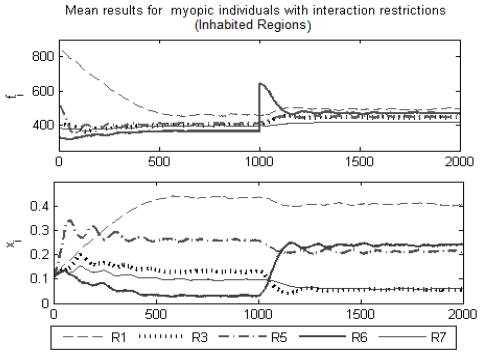

(a) Comp-Time=201.7357




(b) Comp-Time= 98.8256




(c) Comp-Time=67.5012




(d) Comp-Time= 71.0574

Figure 1.8: Results for the experiment No. 3. The mean values for the fitness and the distributions achieved for each case are shown, as well as the computational time for each approach. R1,R3,R5,R6 y R7 are the inhabited regions.

is considerably reduced and it becomes a good option. Both, the proportional and the myopic case, present good results when the interaction restrictions are added (i.e., the agents distribute themselves again according the new profits) and the time of convergence is similar in both cases.

# Chapter 2

# Comparison with other methods

## 2.1 Distributed Gradient (DG)

In [24], [25] and [26] stable models for the control of a group (swarm) of agents are presented through the application of distributed gradient algorithms. These algorithms are based in several simple gradient searches that start in different points in some space, and interact between themselves to achieve some goal. Its goal is to maintain the group cohesion without crashing, and to find the maximum (or minimum) point of some function or profile $\sigma(p_h)$. Each agent $h$ has the dynamics given in Equation (1.6), and its state is determined by its position on the environment $p_h$. The key of this approach is that it takes every agent as a simple gradient search algorithm (individual information), and uses some attraction and repulsion forces between them (social information) to preserve the group and to avoid collisions. If we consider two agents $h$ and $l$, the interaction forces between them are given by

$$R(p_h, p_l) = k_a(p_h - p_l) + k_r \exp\left(\frac{\frac{1}{2}\|p_h - p_l\|^2}{r_s^2}\right)(p_h - p_l) \quad h = 1, ..., P, \quad l = 1, ..., P \tag{2.1}$$

The first term represents an attraction proportional to the distance between agents, the second one describes a repulsion only when they are close, and $k_a$, $k_r$, $r_s$ are positive scalars that represent the attraction intensity, the repulsion magnitude, and the region size around the agent from which it will repel its neighbors, respectively. Finally, the control law for each agent consists in a gradient calculation $\nabla_{p_h}(\sigma(p_h))$, that uses sensing data of $\sigma(\cdot)$, combined with the interaction terms in Equation 2.1. This control rule is described in Equation (2.2), where $k_f$ is the gain of the gradient search.

$$F_h = -k_f \nabla_{p_h}(\sigma(p_h)) - \sum_{l=1, l\neq h}^{P} R(p_h, p_l) \tag{2.2}$$

## 2.2 Ant Colony Optimization (ACO)

In [15] a decentralized adaptive algorithm of task allocation based on a threshold-based model of specialization is proposed. This algorithm is used to allocate tasks in a multiagent system in a flexible and robust way. In [15] the algorithm is applied to a mail company and the objective is to allocate the agents to the various demands that appear in the course of the day, keeping the global demand as low as possible. We can do a direct link between this problem and the general problem that we consider. The demand of each zone is given by the fitness function defined before, and the agents meet a constant quantity of demands $\gamma$ at the point $p_h$ in each iteration. The probability that an agent in region $j$ goes to region $i$ depends on the fitness value of region $j$, the response threshold, and the distance between both regions. The probability that an agent in region $j$ goes to region $i$ is given by,

$$p_i^j = \frac{f_i^2}{f_i^2 + \alpha\theta_{j,i}^2 + \beta d_{j,i}^2} \tag{2.3}$$

where $\theta_{i,j}$ is the response threshold of an $i-$strategist agent to the demand in region

$j$, $d_{j,i}$ is the distance between region $j$ and $i$, and $\alpha$ and $\beta$ are two positive positive coefficients that modulate the influences of $\theta$ and $d$. One important point of this model is that it considers individuals that become specialist in some task when they do it repeatedly. This specialization is controlled through the response threshold, which is updated at every iteration according to,

$$\theta_{i,j} = \begin{cases} \theta_{i,j} - \xi_0 & \text{if } i = j \\ \theta_{i,j} - \xi_1 & \text{if } i \neq j, \text{ and } i \in V_j \\ \theta_{i,j} + \varphi & \text{if } i \neq j, \text{ and } i \notin V_j \end{cases} \tag{2.4}$$

where $V_j$ corresponds to the neighboring regions of region $j$, $\xi_0$ and $\xi_1$ are two learning coefficients corresponding to region $j$ and its neighboring regions respectively ($\xi_0 > xi_1$), and $\varphi$ is a forgetting coefficient for the $i \notin V_j$ regions. The smaller is value of $\theta_{i,j}$, the agent is more specialized in that region.

## 2.3 Genetic Algorithms

A Genetic algorithm is an optimization method that manipulates a string of numbers that represent a solution in a similar way in which chromosomes change in biological evolution. Here, we use the approach developed in [13]. In this case, each chromosome represents the position of an agent in the space and the algorithm evolves these positions in order to achieve the maximum fitness points in the environment. In the selection process, the most profitable chromosomes (agent positions) are chosen to form a mating pool. Consequently, more fit individuals end up with more copies of themselves and they have more influence on the next generation. In the matting pool, two operations are performed: i) crossover: is the interchange of information between parents (interchange of genes after some aleatory point of the chromosome); and ii) mutation:

in this process some genes are randomly changed, this allows the algorithm to avoid local maxima. After that, the next generation of individuals is created. For our case, the next generation represents the new positions of the agents. This process occurs immediately and because of that, we cannot consider the travel time between regions in this approach. If we consider that time, at next iteration the new population will not correspond to the generation created by the algorithm and the new generations are going to follow an erroneous selection process. In the algorithm implemented here, we also use elitism. This makes that the most profitable individual goes to the next generation without changes.

## 2.4 Comparison

To evaluate our proposal we compare it with the swarm intelligence approaches explained in previous sections. In this test, we perform Monte Carlo simulations with $400$ runs and we add a simple dynamic to the environment: we assume that each agent always consumes $\gamma$ units of targets profile at the point $p_h$. We also consider environmental changes, which can depend or not on the agents actions (i.e., agents can modify the toxic substances levels, but also there can be external perturbations to these levels). These changes are given by a Gaussian perturbation similar to the one used in Experiment No. 3 (Section 1.3), which is added to the profile at time $t = 600$. We implemented the individual approach of imitation dynamics with proportional individuals and interaction restrictions. The other algorithms used are: the agent based approach of the imitation dynamics (ID) in Section 1.2; the distributed gradient strategy (DG) in Section 2.1; the ACO zone allocation algorithm in Section 2.2; and the genetic algorithm (GA) in Section 2.3. In all the algorithms but the GA, a travel time proportional to the distances between regions is considered (see why above).

| Algorithm | FL | $\overline{\sigma}$ | Time | $\overline{\sigma(\text{Time})}$ |
|---|---|---|---|---|
| ID | 136.25 | 44.25 | 28.09 | 1.366 |
| GA | 997 | 33.74 | 23.88 | 1.68 |
| ACO | 274.31 | 100.88 | 39.754 | 0.88 |
| DG | 2688 | 148.06 | 28.09 | 1.36 |

Table 2.1: Statistical results for all the algorithms.

We use a $3 \times 3$ grid and a constant population of $100$ agents for all approaches. The size of the grid only have effect in the algorithms that differentiate between regions (ACO and ID). As mentioned before, we select this grid for simplicity in the visualization of the results. In Figure 2.1 the results for all the cases are compared, and the Table 2.1 shows the statistical results for each one. We can see that $ID$ is a competitive algorithm that overcomes the performance of the other approaches, including the $GA$ who is known by its faster convergence. Other important factor is the dependence on the initial conditions. The $DG$ has a strong dependence. The other algorithms have a smaller dependence on initial positions, but the $ACO$ seems to depend more than the $GA$ or the $ID$. In Table 2.1, besides the average values of this data, the convergence time of the algorithms is shown. All times and its standard deviations are in the same ranges. This is evidence that the proposed algorithm does not increase the complexity of the problem. Finally, there is the reaction to environmental changes. We have that all algorithms respond to the disturbance added, with the exception of the $DG$.

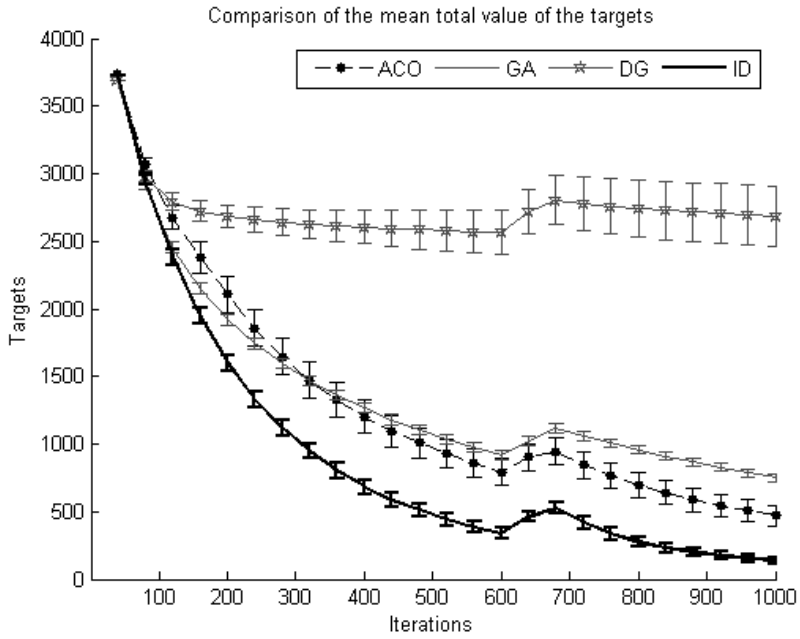
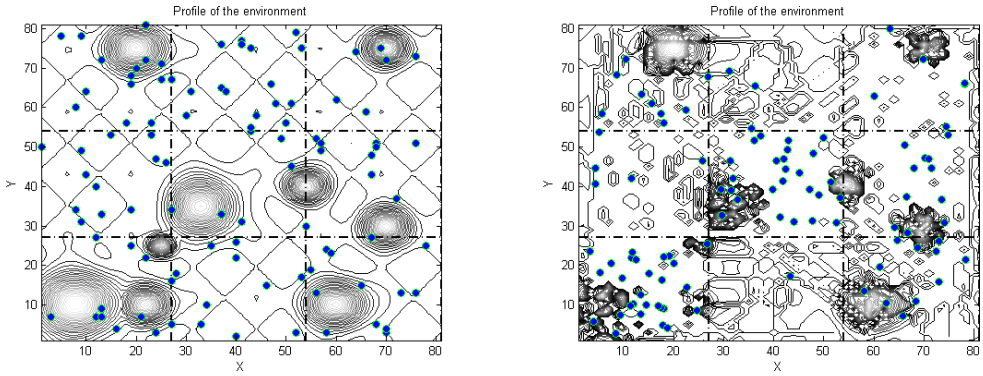



Figure 2.1: Comparison between the evolution of the mean value of total substances for the four methods. The boxes over the curves show the standard deviation related with the initial positions.

# Chapter 3

# Discussion

In this work, we develop an agent based approach for imitation dynamics that uses only local information. Through the experiments realized we can observe that the agents approach follows the predictions made by the continuum model. Besides that, add the interaction restrictions to the dynamics have two important effects: i) it decrease the computational time for the agents; and ii) it makes more difficult the group convergence because we do not have full information. In order to compensate that and avoids agents' maladaptive behaviors, we control the reviewing rates for each agent according its strategy. Here, we have considered two types of individuals: proportional and individual. In general, both have a similar behavior when only local information is used. The selection between them depends on sensing and processing abilities of the agents. The proportional model is more challenging because it processes the information of all neighboring regions at the same time. In the other hand, the myopic case only handles information on one neighboring region. Other parameter that increases the complexity of the problem is the size of the grid. The grid complexity depends on the sensor capacities of the agents and the type of the environment with we are dealing. If we increase the grid complexity, we have to increase the restriction to the reviewing rates




(a) Iteration=1 (b) Iteration=200

(c) Iteration=601 (d) Iteration=800

Figure 3.1: Evolution of the environment defined by the multimodal Gaussian profile. The agents use the individual based approach of imitation dynamics. Lighter lines represent the highest targets levels.

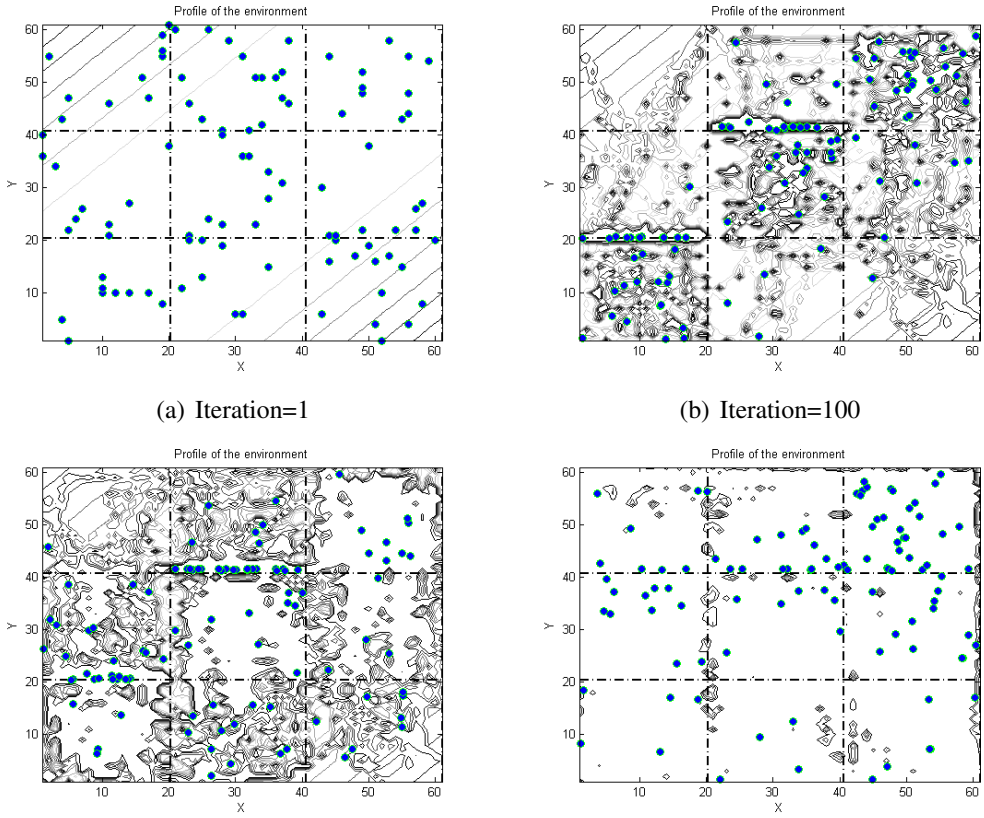
too. With respect to the type of individuals used, the computational time required by the myopic individuals is almost the same with different grids. Instead, the computational time required by the proportional ones increase with the grid complexity. This happens because when the grid complexity is increased, the number of regions with $8$ neighboring regions grows.

Figures 3.1 and 3.2 show the evolution of the environment for the two profile functions mentioned before: the multimodal Gaussian profile and the Rosenbrock function, respectively. We can see how the agents go first to the most profitable regions and they meet their demands. These regions correspond to $j = 1, 3, 5, 7$, for the Gaussian

profile, and to $j = 1, 5, 9$ for the Rosenbrock one. After the finesses in these regions decrease, the agents start to meet the target demands in other regions (parts (c) and (d) on both figures). In Figure 3.1(c), we can see the disturbance that is added to the to see how the agents reallocate. This disturbance changes the fitness on region $6$. In Figure 3.1(d) we can see how the agents reallocate themselves in that region, and meet its demand. The multimodal Gaussian profile is a more interesting testing because represent more diverse scenarios. If we look the Rosenbrock function, we can see that the total value of resources in the different regions is not so difficult to discriminate. We have the three most profitable regions at the diagonal and the level of substances decrease symmetrically around it. This kind of profile can be followed for an algorithm like the distributed gradient easily. However, the distributed gradient can not follow the multimodal Gaussian profile.

Our approach is compared with other swarm intelligence algorithms and it is shown that it overcomes the other approaches. The gradient distributed algorithm has a strong dependence on the initial positions and it gets stock when the agents modify the environment. The genetic algorithm needs a high crossover and mutation probabilities to achieve the proposed objective. Finally, the ACO approach gives an acceptable behavior of the group. However, it needs that the agents be able to know the data of any region in the environment and it does not always go to the most profitable regions first. Because of this, its convergence is slower than the imitation dynamics case.

If we look the results of the experiments in Section 1.3, we can associate the distribution achieved with an important concept on behavioral ecology. The ideal free distribution (IFD), originally introduced in [30], characterizes an equilibrium distribution where all animals achieve equal fitness, and any individual can increase its fitness by an unilateral change of strategy. If we organize the regions according to its quality, then the agents will reach an equilibrium point that corresponds to the IFD, where all




(a) Iteration=1

(b) Iteration=100

(c) Iteration=301

(d) Iteration=800

Figure 3.2: Evolution of the environment defined by the Rosenbrock function. The agents use the individual based approach of imitation dynamics. Lighter lines represent the highest targets levels.

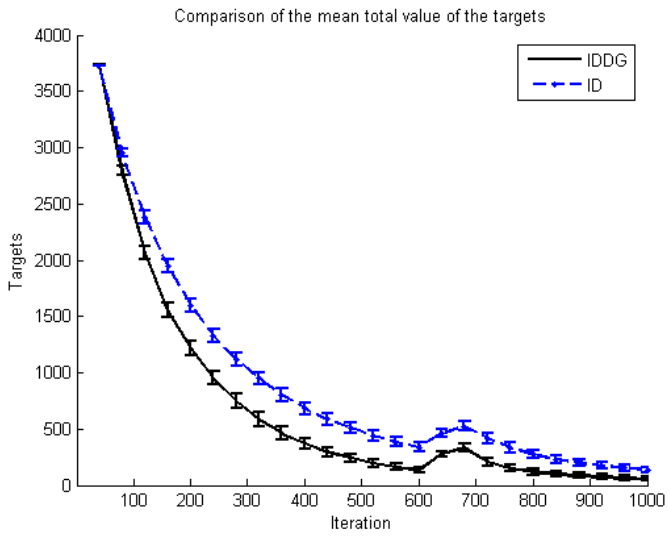



Figure 3.3: Comparison between imitation dynamics and a combination this algorithm and the distributed gradient one.

have the same fitness and the first most profitable regions are occupied. This kind of distribution is achieved by the agent based on imitation dynamics, even when some original assumptions are relaxed, such as the no cost travel between regions or the full knowledge of the environment.

The algorithm presented here has a very good performance, but it has some drawbacks. It does not take into account the positions of other agents, and consequently, it is possible that the random selection process generates collisions between the agents. We consider that a way to overcome this problem is to combine the imitation dynamics with the interactions used by distributed gradient algorithms. Other advantage of implementing this combination is that the $DG$ improves the velocity of the local searches in each region (until now this is done in a random way). If $ID$ assigns each agent to a profitability region, then the $DG$ is close to some maximum point and its performance is good. When we combine the two approaches, the goal is achieved more quickly. Figure 3.3 shows the results for both approaches.

# Conclusion

In this work we study how to apply bio-inspired learning techniques to a group of agents, in order to solve a general problem in which we should hold some toxic substances at their minimum levels. In this problem, the agents should be capable to distribute themselves through the environment and to find the more important focus of the substance profile. We center our study on bio-inspired approaches because they have a great potential in multi-agent applications, because some of these approaches use indirect communication systems and simple individual actions. The proposed algorithm is an individual based approach of the imitation dynamics used in evolutionary game theory. This technique achieves distributions in which all agents obtain similar benefits and they can attack the demand in parallel. Through some simulations tests, it is shown that our algorithm overcomes other swarm intelligence techniques, such as ACO, genetic algorithms, and distributed gradient searches. Other important contribution of the work is to help to understand how optimal distributions can be achieved by a group driven only by local rules. Specifically, we find that to control the frequency at which agents review their strategy helps to control the maladaptive behavior occasioned by mass effects, and that profitable distributions can be achieved by non-ideal individuals in different environmental situations. This fact allows us to overcome excessive computational requirements, and to implement simpler and less expensive solutions. In the future, it would be useful to analyze the mathematical model with the restrictions and

prove the algorithm in more complex scenarios.

# Bibliography

[1] P. Stone and M. M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Autonomous Robots*, vol. 8, no. 3, pp. 345–383, 2000.

[2] P. Brazdil, M. Gams, S. Sian, L. Torgo, and W. van de Velde, "Learning in distributed systems and multi-agent environments," in *EWSL-91: Proceedings of the European working session on learning on Machine learning*. New York, NY, USA: Springer-Verlag New York, Inc., 1991, pp. 412–423.

[3] S. Nakrani and C. Tovey, "From honeybees to internet servers: biomimicry for distributed management of internet hosting centers," *Bioinspiration & Biomimetics*, vol. 2, no. 4, pp. S182–S197, 2007.

[4] D. Zhang, G. Xie, J. Yu, and L. Wang, "Adaptive task assignment for multiple mobile robots via swarm intelligence approach," *Robot. Auton. Syst.*, vol. 55, no. 7, pp. 572–588, 2007.

[5] K. Sekiyama, Y. Kubo, and Y. Ohashi, "Geometrical analysis and design of collective strategy formation," *Robotics, Intelligent Systems and Signal Processing, 2003. Proceedings. 2003 IEEE International Conference on*, vol. 1, pp. 331–338 vol.1, Oct. 2003.

[6] F. Camci, "Comparison of genetic and binary particle swarm optimization algorithms on system maintenance scheduling using prognostics information," *Engineering Optimization*, vol. 49, no. 2, pp. 119–136, February 2009.

[7] U. Ozgue and B. Serol, "Comparison of genetic algorithm and particle swarm optimization for bicriteria permutation flowshop scheduling problem," *International Journal of Computational Intelligence Research*, vol. 4, no. 2, pp. 159–175, 2008.

[8] J. Finke and K. Passino, "Stable emergent heterogeneous agent distributions in noisy environments," *American Control Conference, 2006*, pp. 6 pp.–, June 2006.

[9] L. Navarro-Serment, R. Grabowski, C. Paredis, and P. Khosla, "Millibots," *Robotics and Automation Magazine, IEEE*, vol. 9, no. 4, pp. 31–40, Dec 2002.

[10] S. Kalantar and U. R. Zimmer, "Distributed shape control of homogeneous swarms of autonomous underwater vehicles," *Auton. Robots*, vol. 22, no. 1, pp. 37–53, 2007.

[11] Y. Altshuler, A. Bruckstein, and I. Wagner, "Swarm robotics for a dynamic cleaning problem," *Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE*, pp. 209–216, June 2005.

[12] Y. Jin, Y. Liao, A. Minai, and M. Polycarpou, "Balancing search and target response in cooperative unmanned aerial vehicle (uav) teams," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 36, no. 3, pp. 571–587, June 2005.

[13] K. Passino, *Biomimicry for Optimization, Control and Automation*. London: Springer-Verlag, 2005.

[14] L.-A. Giraldeau and T. Caraco, *Social Foraging Theory*. Princeton University Press, May 2000.

[15] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*. New York, NY, USA: Oxford University Press, Inc., 1999.

[16] R. C. Eberhart, Y. Shi, and J. Kennedy, *Swarm Intelligence*, ser. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann, March 2001.

[17] J. W. Weibull, *Evolutionary Game Theory*. The MIT Press, 1997.

[18] R. Cressman and V. Krivan, "Migration dynamics for the ideal free distribution." *The American Naturalist*, vol. 168, no. 3, pp. 384–397, September 2006.

[19] N. Quijano and K. Passino, "The ideal free distribution: Theory and engineering application," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 1, pp. 154–165, Feb. 2007.

[20] ——, "Honey bee social foraging algorithms for resource allocation," *American Control Conference, 2007. ACC '07*, pp. 3383–3394, July 2007.

[21] A. Pantoja and N. Quijano, "Modeling and analysis for a temperature system based on resource dynamics and the ideal free distribution," *American Control Conference, 2008*, pp. 3390–3395, June 2008.

[22] B. J. Moore, J. Finke, and K. M. Passino, "Optimal allocation of heterogeneous resources in cooperative control scenarios," *Automatica*, vol. 45, no. 3, pp. 711 – 715, 2009.

[23] W. Schnedler, "Traits, imitation and evolutionary dynamics," Department of Economics, University of Bristol, UK, The Centre for Market and Public Organisation 03/081, Jul 2003.

[24] G. g. Rigatos, "Distributed gradient and particle swarm optimization for multi-robot motion planning," *Robotica*, vol. 26, no. 3, pp. 357–370, 2008.

[25] V. Gazi and K. Passino, "Stability analysis of social foraging swarms," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 34, no. 1, pp. 539–557, Feb. 2004.

[26] Y. Liu and K. Passino, "Stable social foraging swarms in a noisy environment," *Automatic Control, IEEE Transactions on*, vol. 49, no. 1, pp. 30–44, Jan. 2004.

[27] J. Hofbauer and K. Sigmund, *Evolutionary Games and Population Dynamics*. Cambridge University Press, May 1998.

[28] S. K.H., "Why imitate, and if so, how? a boundedly rational approach to multi-armed bandits," *Journal of Economic Theory*, vol. 78, pp. 130–156(27), January 1998.

[29] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information Processing Letters*, vol. 85, no. 6, pp. 317 – 325, 2003.

[30] P. Taylor and L. Jonker, "Evolutionary stable strategies and game dynamics," *Mathematical Biosciences*, vol. 16, pp. 76–83, 1978.