Evolution of Cooperation with Partner Choice in Collective Adaptive Systems

Paul Ecoffet

A thesis presented for the degree of
Doctor of Philosophy in Computer Science

ISIR - Institut des Systèmes Intelligents et de Robotique
Sorbonne Université
Paris, France
April 13, 2021

Nicolas Bredeche - PU - ISIR, Sorbonne Université	Advisor
Jean-Baptiste André - CR CNRS - IJN, ENS	Co-Advisor
Eliseo Ferrante - Associate Professor - VU Amsterdam	Reviewer
Guillaume Achaz - PU - MNHN, Paris-VII	Reviewer
Silvia De Monte - CR CNRS - IBENS, ENS	Examiner
Nicolas Maudet - PU - LIP6, Sorbonne Université	Examiner

Dedication

To my father, Franck Ecoffet

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or processional qualification except as specified.

Contents

List of Figures

Introduction

The evolution of cooperation between genetically unrelated individuals is an apparent paradox from an evolutionary point of view. Indeed, all individuals in nature should be self-interested. Through the process of evolution, only the *fittest* individuals propagate. Helping another individual is thus a waste of time and resources, as it does not help the actor's fitness at first sight. How come, then, that some individuals act in cooperative ways, helping, foraging, hunting, to the benefit of genetically unrelated others? Several mechanisms have been unravelled that can explain how these cooperative behaviours could have been selected through evolution. In these mechanisms, the recipient of the cooperation rewards the actor, eventually fulfilling the self-centred interest of the actor. Such *reciprocal* behaviour creates an incentive for individuals to act cooperatively. However, reciprocal cooperative behaviours are relatively rare in nature. The conditions necessary for their evolution are very restrictive and rarely met.

The cooperation problem is also a complicated and challenging issue in artificial systems composed of several agents. These systems can be networks of computers or even swarms of robots. Aligning the behaviour of the agents with the global goal of the artificial system is a complex task. Agents can behave in deleterious ways to the fulfilment of the global goal. They can worsen others' efficiency by maximising theirs. Fine-tuning agent behaviours by hand is a complicated and daunting task. Machine learning methods can be used to overcome this intricate behavioural design. However, machine learning techniques also suffer from the same caveats as expert behavioural design. Machine learning algorithms require an objective function to optimise. If the objective function at the agent level does not align with the global objective function of the system, the agents' behaviours can eventually be detrimental to the system. The use of machine learning thus shifts the problem from the design of an optimal behaviour to the design of an efficient objective function. By being able to design objective functions at the individual level that are aligned with the overall goal, it would be possible to resolve this dilemma of local learning for collective effectiveness. Being able

to enforce the agents to *cooperate* in the fulfilment of each other's tasks, instead of maximising their *own* efficiency, can solve the dilemma of collective effectiveness.

The objective of this thesis is twofold: First, the goal is to study the settings observed in biology that have led to the development of cooperation in living organisms and to transpose these settings to artificial setups. These settings are composed of the agents' environments, but also of their observation and behavioural capacities. In this thesis, we mainly focus on the partner choice mechanism, a specific form of reciprocal behavior, as a way to enforce cooperative behaviour. Second, the design of cooperative artificial systems allows a deep understanding of the environmental constraints that weigh on the emergence of cooperation in the living. These studies, therefore, expand the results in theoretical biology and evolutionary game theory.

In Chapter 1, we will present the evolutionary paradox of cooperation. We will first recall in its broad outline the mechanism of evolution by natural selection. We will explain why cooperation does not seem viable in this scheme at first sight. We will then explain how cooperative behaviours can actually evolve and be evolutionarily stable, by using the framework of evolutionary game theory and focusing in particular on the evolution of cooperation by partner choice. We will detail several theoretical models of partner choice that uncover many prerequisites of this mechanism, and present several examples of cooperation in the living world where partner choice plays a role.

Chapter 1 will also aim at showing the connection between evolutionary biology and robotics. Evolutionary robotics draws on the results of evolutionary biology to design algorithms based on evolutionary processes. We will detail the functioning of evolutionary algorithms and their applications in swarm robotics for the acquisition of cooperative behaviour. Furthermore, we will show how evolutionary robotics is in itself a method for understanding the emergence of cooperation in nature.

Most works on partner choice show that the easier it is for an individual to compare various cooperation opportunities, i.e. to compare various potential partners, the more effective partner choice is as a mechanism to foster cooperation. However, none of the models developed so far in the literature consider explicitly the fact that individuals need to find environmental resources (e.g., a prey to catch, a tuber to dig up, etc.) in order to interact with each other. In Chapter 2, we introduce explicitly this ecological component in a partner choice model. In this framework, a cooperation opportunity does not only consist of an available partner, it consists of the conjunction of a partner and a resource to exploit with that partner. As a result, the efficiency of partner choice as a mechanism to foster cooperation does not only depend

on the availability of potential partners. It depends on two factors: (i) the quality of the environment in terms of available resources, and (ii) population density. We show that the efficiency of partner choice, and hence the eventual evolution of cooperation, depends on a complex interaction between these two factors.

When environment quality, i.e. the density of resource patches in the environment, is low relative to population density, resource patches are saturated, hence individuals do not need to compete to attract partners on their patch, and partner choice is thus inefficient to foster cooperation. When population density is too low, on the other hand, partner choice is also inefficient regardless of the density of resource patches, because individuals can never find enough partners to compare them with each other. Only when population density is at an intermediate level, approximately matching the density of resources, can partner choice operate and lead to the evolution of cooperation.

The model developed in Chapter 2 is non spatialized. In Chapter 3, we extend this approach in a spatialized model where agents have to navigate in their environment to find a partner and/or a resource patch. We study how spatialization impacts the evolutionary dynamics of cooperation by partner choice in the context of swarm robotics. By using a swarm robotics setup, we gain a two-fold contribution. First, swarm robotics simulation offers a natural extension towards spatial environment with respect to previous partner choice models. Second, partner choice may provide an efficient mechanism to be used within a collection of learning robots, whenever each robot is considered as independent from the others with respect to its own objective function.

The model built in Chapter 3 thus shows the possibility for agents to learn (through a genetic algorithm) to act cooperatively in a swarm robotics framework. However, the learning algorithm used in Chapter 3 is not completely decentralised. Decentralised and online learning is necessary in order to design robust and adaptive swarms. In Chapter 4, we thus focus on designing efficient setups and algorithms for learning cooperative behaviour in swarm robotics and explore fully decentralised algorithms in a very simplified cooperative task setup with particularly rare significant events. In this chapter, we compare a state of the art deep reinforcement learning algorithm (PPO) to an evolutionary strategy algorithm (CMA-ES). We compare their tolerance to the rarity of significant events. This work shows the immense difficulty that deep reinforcement learning algorithms have in managing great sparsity of interesting event, which ESes do not bear. These results make it possible to better identify the constraints that learning algorithms need to resolve in order for the robots in a swarm to act cooperatively through

decentralised and online learning.

Chapter 1

Evolution of Cooperation and Partner Choice

1.1 The Evolution of Cooperation

1.1.1 Evolutionary approaches to behaviour

Evolutionary biology studies the emergence and maintenance of morphological and behavioural traits in life (West et al., 2007a). Each individual has a number of traits that can be either advantageous or disadvantageous to the individual compared to its peers. A trait is said to be *advantageous* if it allows the individual to have *a greater reproductive success* than its peers. This means that the individual has a greater ability to survive and reproduce if it has this trait. This ability is called the individual's fitness.

If an advantageous trait is transmissible by reproduction, since the individual owning this trait reproduces more than other members of the population, the more adapted trait is increasingly common in the population. The offspring of this individual are also more adapted than the offspring of other individuals, and therefore have greater reproductive success, and so on. The frequency of the new trait in the population increases over generations to the point where the trait is virtually owned by the entire population. The trait has become *fixed* in the population. This whole process is the mechanism of evolution by natural selection.

Thus, through natural selection, the most suitable traits spread and become fixed in the population. Here, we have taken the simple case of a single trait in a stationary environment. The fixation of traits can be much more complex. The traits that define an individual may interact with each other, or the environment in which the population evolves may change. This environment may change either exogenously to the process of individuals'

adaptation, or because of the changes in individuals' behaviour and morphology.

Natural selection is the first ingredient in the mechanisms of evolution. The second element is variation. We have previously postulated that an individual has a different trait from other members of its population and we have made the implicit assumption of perfect transmission of traits from parent to offspring. This perfect transmission was a simplification for explanatory purposes. Actually, during reproduction, there is a probability that the transmitted trait is slightly altered. This alteration may have marginal consequences — the trait is more or less expressed, or it may even have significant consequences, such as completely changing the nature of the trait. This new version of the trait may or may not be adapted, and therefore undergo the same selection process as described above.

Thus, the variation mechanism introduces new versions of the traits into the population, while the selection process keeps the most suitable traits that will fix themselves in the population at the expense of the less suitable ones. This process, evolution, leads to a process of optimisation. With each generation, the traits that maximise the survival and reproduction of individuals are propagated.

Let us now look in more detail at the transmission of traits. It is not the traits themselves that are actually transmitted between individuals, but the substrate that encodes them: the genes. A gene is an element encoding a trait. A variation in the code contained in the gene potentially leads to a variation in its expression, the trait. Several individuals may possess the same gene, which will code for the same trait in each individual. Of course, most of the traits we observe in living organisms are complex and are due to the expression of several interacting genes. In this context, it is no longer the individual that is at the centre of evolution, but the genes that compose it. A gene exists and reproduces. It exists in many copies. The most adapted genes — those that best enable the survival and reproduction of the individuals who carry them — become fixed in the population, to the detriment of their less adapted alternatives.

1.1.2 The Problem of Cooperation

In evolutionary biology, cooperation is defined as acting to benefit another individual in terms of reproductive success (West et al., 2007a). Cooperation is a very paradoxical behaviour in evolutionary biology. Indeed, how can such behaviour be adaptive? A cooperating individual spends time and energy to increase the fitness of another. This time and energy could as well been spent on increasing the fitness of the cooperator *itself*. It is easy to understand that

being the recipient of cooperation is adaptive and that this behaviour can be maintained by natural selection. Nevertheless, it is confusing that being the actor of cooperation can be adaptive.

However, many cases of cooperation are observed in the living, from the microbial to the mammalian level (Clutton-Brock, 2009; Krams et al., 2008; Schroeder et al., 2014; Simms & Lee Taylor, 2002; Wilkinson et al., 2016). First of all, one of the most striking examples could be eusocial insects, which include ants or bees. In these species, members of entire colonies coordinate and work together to ensure the reproductive success of their queens. It's a form of cooperation that we'll discuss in the section 1.1.3 *Kin selection and indirect fitness benefits*. Cooperative behaviour is also observed in vampire bats (Figure 1.1a; Wilkinson et al., 2016). If a bat has failed to feed itself, another vampire bat comes to its rescue by regurgitating its meal to feed the hungry bat. The bat that regurgitates its meal cooperates. It pays a cost (it puts itself in danger) to help another individual. Pied flycatchers are also cooperators (Figure 1.1b; Krams et al., 2008). These birds are the prey of owls. When an owl hunts a pied flycatcher, the pied flycatcher calls for help. The flycatchers in the vicinity help the hunted bird by attacking the owl in groups. The flycatchers who come to help the prey risk their lives to save another individual. Other examples can be found in the vegetable-rhizobia mutualism (Figure 1.1c; Simms and Lee Taylor, 2002), which we will detail later. Finally, the human species is also characterized by a tremendous level of cooperation between individuals (Schroeder et al., 2014). Humans possess complex social structures, produce their resources together, exchange them, and even donate or work voluntarily to charitable organizations (Figure 1.1d). Collective and cooperative behaviours are numerous in the living world.

However, as stated before, these cooperative behaviours are apparently paradoxical. For example, as a first approximation, a pied flycatcher that would not attack the predator of one of its partners should have a better fitness by not endangering its own life. How then can we explain that these behaviours have not been counter-selected? This question is much studied in evolutionary biology. Numerous studies have identified the different mechanisms that make cooperative behaviours adaptive (Hamilton, 1964; Sachs et al., 2004; Schroeder et al., 2014; Trivers, 1971; West et al., 2007a). These mechanisms are detailed in the next sections. We will first briefly discuss cooperation between genetically related individuals, then move on to the mechanisms of cooperation between non-related individuals, particularly the mechanisms of partner choice, which are the focus of this thesis.

In this thesis, we call cooperation the act of paying a cost to give a gain to another in terms of selective value. The individual who pays the cost will be called the *actor*, the individual who receives the payoff of the action

(a) Vampire Bats

(b) Pied Flycatcher

(c) Legumes and rhizobia

(d) Humans

Figure 1.1: Vampire bats, legumes and rhizobia, pied flycatchers and humans cooperate, investing time and resources for other individuals with no apparent benefits.

will be called the *recipient*, consistent with West et al. (2007a). During an interaction between two individuals, the individuals can be either only actor or recipient, or both simultaneously. Similarly, in future interactions, the roles of actor and recipient may be reversed. Finally, there may be several actors and recipients in the same interaction.

1.1.3 Kin selection and indirect fitness benefits

Kin selection is a first evolutionary mechanism that explains that cooperative behaviours can be adaptive (Hamilton, 1964). Indeed, the relevant unit for the propagation of a trait is the gene in all of its copies, not the individual by itself (see section 1.1.1). A gene can increase its propagation by improving the fitness of the individual carrying it, but also by increasing the fitness of other individuals carrying the same gene. Thus, a gene that helps its carrier's siblings or offspring increases the fitness of related individuals. It is likely that these relatives also carry a copy of this gene since they are from the same parents. Therefore, this gene indirectly increases *its* fitness. Thus, a mutant gene that helps relatives of an individual compared to a resident gene that does not help them is favoured by natural selection.

However, kin selection can only explain the existence of a subset of the cooperative behaviours observed in the living world, those expressed towards genetically related individuals. How could cooperative behaviours between unrelated individuals, as observed in humans or vampire bats, for example, or cooperative behaviours between individuals of different species, as observed in symbioses, have been favoured by natural selection?

1.1.4 Mutualism

For cooperation between genetically unrelated individuals to be evolutionarily stable, the actor must itself receive a benefit from its cooperative behaviour. This benefit can be obtained if others respond to its cooperative behaviour later by cooperating back (Trivers, 1971).

This mechanism is called conditional cooperation or reciprocity. There are two main families of conditional cooperation mechanisms: *Partner Control* (also called Partner Fidelity Feedback) and *Partner Choice* (Noë, 2006; Sachs et al., 2004). In Partner Control, the recipient of an interaction adjusts its behaviour towards the same actor and continues to interact with it. Reciprocity can be either positive, i.e. the recipient cooperates with the actor in response to the cooperation, or negative, i.e. the recipient punishes the actor if the actor does not cooperate. In both cases, it is in the actor's interest to cooperate, since this maximizes its gain from the recipient's response.

Partner Control behaviours can be implemented very easily and are particularly robust, as shown in Axelrod and Hamilton (1981). Thus, in a cooperative situation which can be modelled as a Prisoner's Dilemma (detailed in section 1.2), with repeated interactions, the tit-for-tat behaviour is a robust and straightforward reciprocity strategy. The tit-for-tat strategy is an optimistic imitation strategy. It consists of always starting any interaction by cooperating and then imitating its partner during the following time steps. Thus two individuals playing this strategy will cooperate at every time step of the interaction. On the other hand, when an individual playing tit-for-tat interacts with a cheater, it is exploited only once, and then it stops cooperating.

Tit-for-tat behaviours have been observed in pied flycatchers (Krams et al., 2008), who only come to defend partners who have defended them before, and refuse to help those who have not come to help them when they needed it. They are also seen in wild vervet monkeys for grooming (Fruteau et al., 2009).

1.1.5 Partner Choice and Biological Markets

In partner choice, on the other hand, individuals do not only adjust their cooperation with a given partner according to his past action. They *choose* their partner according to their past action. Since all individuals in the population seek to be with the best possible partner and not all can meet their demand, there is a *biological market* of partners (Noë & Hammerstein, 1994).

We observe partner choice processes in a wide variety of living systems, from the cleaner fishes (Bshary & Grutter, 2002) to the legumes-rhizobia mutualisms (Simms & Lee Taylor, 2002) that we will develop further. In the human species, partner choice has likely played a prominent role in the evolution of cooperative behaviour (Barclay, 2013; Barclay & Willer, 2007; Debove, André, et al., 2015).

Thus, partner choice allows the appearance and maintenance of cooperation. To understand this, let us no longer focus on the actor of cooperation, but on the recipient. In a collective task, it is always relevant for the recipient to interact with the best possible actor, i.e. the actor who will enable it to obtain the biggest gain. Since in order to perform this collective task, the actors need the recipients, it is then in the interest of the actors to be as cooperative as possible to be picked. This pressure is all the stronger if the number of actors is particularly large for the number of recipients. The actors and recipients are in a supply and demand setup, which can be studied in the form of a market (Noë & Hammerstein, 1994).

Let us consider a population of individuals who are looking to interact with the best possible partner. In this population, a mutant appears who cooperates more than the others. The other individuals will particularly desire to interact with this mutant. The mutant will therefore be involved in a lot of interactions and obtain many gains. As a result, the mutant can be picky. It will be able to refuse interactions with the least efficient individuals in order to choose the most efficient partners. There is therefore an "*assortative matching*". That is to say that individuals will be matched according to their performance. The best performing individuals will be able to afford to be picky and will end up being paired with other well performing individuals, and vice versa the worst-performing individuals will pair up together(Geoffroy et al., 2018). Therefore, the best performing individuals who interact together will receive benefits from their high level of cooperation. That is, assortative matching generates a selective pressure in favor of cooperation.

For example, Bshary and Grutter (2002) shows that cleaners fishes and their clients cooperate in a market structure. Cleaner fishes are small fishes that eat the parasites present on "client" fishes. Cleaner fishes have "cleaning stations". They always stay in the same area. When clients want to be cleaned, they go to these stations. The clients can select which station they go to. Depending on the supply of cleaners and the demand of clients, the market can achieve different balances in favour of the clients or the cleaners: If there are fewer stations than necessary to meet the total demand of the clients, then the clients are in an unfavourable situation, it is difficult for them to access a station. The cleaners take advantage of this situation: They allow themselves not only to eat the parasites present on the clients but also to eat their mucus tissues, which are very nutritious for the cleaners. The cleaners are cheating. Since the clients have no other options, they can only comply. On the contrary, when there are more stations than necessary to accommodate all the clients, there is more supply than demand, and the clients are in an advantageous situation. The cleaner fishes do not eat the mucous membranes of the clients, because if the clients are not satisfied with the service provided, they can go to another station next time.

The mechanism is similar in the legume-rhizobia mutualism (Simms & Lee Taylor, 2002). Legumes need nitrogen that they cannot capture from the air. Many bacteria in the soil, the rhizobia, release nitrogen elements that the plant can capture. The rhizobia also need the help from the legume because they consume carbon elements that the plants produce. Thus, legumes create in their roots nodules that host and supply carbon elements to the bacteria. Inefficient nodules, where the bacteria produce little nitrogen, are destroyed and deprived of carbon, while efficient nodules are maintained and supplied with carbon. There is a market effect, and partner choice develops. The

plant hosts and provides resources only to the bacteria that offer nitrogen in exchange.

Note that partner choice can be implemented in many different ways, varying in complexity and efficiency. For example, partner choice can be achieved through direct information. The individual looking for a partner, the chooser, uses its knowledge of the different partners (Aktipis, 2004, 2011; Debove, Baumard, et al., 2015; McNamara et al., 2008). If the chooser uses only the information from the current partner, this partner choice is called partner switching. It can be worded as a simple rule: If the current partner cooperates, the chooser stays with it; otherwise it switches to a new partner at random (Aktipis, 2011; Bshary & Grutter, 2005). The chooser can also use a memory of all past interactions with its partners to pick the best partner available directly. Partner choice can also be made through *indirect* information. The chooser picks a partner based on the partners' past interactions with other individuals. This knowledge can come from direct observation or reported information from other individuals.

1.1.6 Why isn't cooperation everywhere?

Although we wondered at the beginning of this chapter how cooperation could evolve, after studying the different mechanisms that could support it, it is now the opposite question that emerges. Why is reciprocal cooperation relatively rare in nature? Indeed, all examples of reciprocity in animals are contested (reviewed in part in Carter, 2014), and yet partner choice is an incredibly powerful mechanism in Humans (Barclay, 2013; Barclay & Willer, 2007; Debove, André, et al., 2015). What factors might prevent the emergence of reciprocity?

First of all, a substantial problem is the bootstrapping issue (André, 2014). While it is easy to understand how reciprocity mechanisms can maintain cooperative behaviours, it is more complicated to explain how this mechanism can appear by itself. Indeed, reciprocity — be it in the partner control or in the partner choice version — requires two mutually dependent traits, both unstable by themselves: it requires that (i) the actor can cooperate and that (ii) the recipient can recognize and respond to an act of cooperation.

Without the simultaneous presence of these two traits, reciprocity cannot take place, and cooperation is not evolutionarily stable. Indeed, the ability to distinguish a cooperative partner from a cheating partner only makes sense if there are both cooperative and non-cooperative individuals in the recipient's vicinity. If the individual's neighbourhood consists only of cheaters (or only of cooperators), then there is no benefit in maintaining a complex system of cooperator recognition.

Similarly, cooperative behaviours have no reason to be maintained by natural selection if there is no individual able to recognize and respond conditionally to them. Indeed, cooperation is evolutionarily stable only if paying a short-term cost makes it possible to change the recipient's future behaviour. If the selected recipient does not have the competence to distinguish a cooperator from a cheater, its behaviour cannot change. Therefore, there cannot be any interest in cooperating.

These two traits, which are both complex and different, can only be favored together. However, it is extremely improbable that these two traits will appear at the same time in a population. One solution to overcome this gap is that either of these behaviours already existed at least partially in the population for other reasons. For example, one hypothesis to allow the emergence of cooperation between unrelated individuals is that the cooperation implemented by kin-selection can sometimes be applied between non-kin by misfiring. Another hypothesis is based on the role of byproduct cooperation as a triggering factor (André, 2015).

Beyond this bootstrapping problem, however, other constraints influence the evolution of cooperation by partner choice. Partner choice requires the presence of numerous and accessible outside options so that comparing different partners is viable (Chade et al., 2017; Debove, Baumard, et al., 2015; Raihani & Bshary, 2011). If it is too costly for an individual to find a better partner compared to the gain obtained with their current partner, it is not advantageous to be choosy. We will develop this point further in Chapter 2 and show that it could play an important role in the phylogenetic distribution of cooperation. In Chapter 3, we explore the possibility of the emergence of cooperative behaviours by partner choice in pseudo-realistic environments, studying the impact of these emergence issues.

1.2 Models for the Evolution of Cooperation

To study the adaptability of behaviours, biologists use several theoretical tools. Imported from economics, Game Theory enables the construction of scenarios that model the dilemmas faced by individuals. This framework makes it possible to study the behaviours of individuals who interact with each other and seek to maximize their gain. This framework allows the study of the evolution of social behaviours in biology (André & Day, 2007; Axelrod & Hamilton, 1981; Dittami, 2001; Robson, 1990).

The players' objective is to maximize their gains by choosing the best possible action. Furthermore, the best possible action to play also depends on the actions chosen by other players. When all players play the best possible

action knowing what other players are playing, the game reaches a *Nash equilibrium*. In this case, there is no player i such as if all other players keep the a_{-i} action set, there is an action $b_i \in \mathcal{A}_i, b_i \neq a_i$ such that $P_i^{a^{-i}}(b_i) \geq P_i^{a_{-i}}(a_i)$. Multiple Nash equilibria are possible for the same game, and a Nash equilibrium may be sub-optimal, i.e. at least one other set of actions could be played leading to a greater or equal payoff for all players.

By transferring vocabulary from game theory to evolutionary biology, it is possible to use the mathematical framework of game theory to study the evolution of behaviours. Thus, by using for Payoff the fitness function of individuals, and by stating that the agents' genome encodes their strategies, i.e. the actions they will play, it is then possible to determine which strategies are propagated in the population. A strategy s can then be evolutionary stable (ESS). That is, if a resident population implements the evolutionary stable strategy s, then for any appearance of a mutant with a strategy $s' \neq s$, then $P(s, s) > P(s', s)$ [1](Maynard Smith, 1974). An ESS is necessarily a Nash equilibrium.

The problem of cooperation can be modelled within the framework of game theory. The most classic model for studying cooperation in game theory is the Prisoner's Dilemma. In this game, two players interact together. They both possess the same action repertoire: They can either *cooperate* or *defect*. Based on the actions of both agents, each receives a payoff. Below is an example of a Prisoner's Dilemma with its game matrix (see Table 1.1). Different versions of the Prisoner's Dilemma exist, varying each agent's Payoff based on their actions by changing the values $b > c > 0$.

a_1 \ a_2	Cooperate	Defect
Cooperate	b - c b - c	b -c
Defect	-c b	0 0

Table 1.1: Game matrix of a Prisoner's Dilemma, $b > c > 0$

This dilemma captures the problem of cooperation because of the conflict between the individual and the collective interest. For each agent, in order to maximize its payoff, it is better to play *defect* than to play *cooperate*. Indeed, no matter what their partner does, the action *defect* will always bring them a better payoff. However, if they both *defect*, they both get less payoff than

[1]there is a borderline case that if $P(s, s) = P(s', s)$ and $P(s, s') > P(s', s')$, s is also ESS

if they both *cooperate*. There is a sub-optimal Nash equilibrium for the game where both players *defect*. In the case of an unrepeated game, it is rationally more viable for both players to *defect*. Moreover, this Nash equilibrium is also an ESS. In this case, the collective (and also individual) efficiency is sub-optimal.

However, if the game is repeated, the nature of the game changes. If the same partners play together several times, an agent can induce their partner to cooperate, thanks to the reciprocity strategies presented earlier, here *Partner control*. Thus, Axelrod and Hamilton (1981) have shown in the context of the Prisoner's Dilemma that the most stable strategy among a set of several dozen strategies in a tournament with replication was tit-for-tat. Moreover, this game strategy is remarkably simple to implement. It requires almost no memory and has a single conditional branch and can thus emerge very easily by evolution.

Tit-for-tat is a stable strategy[2] amongst many other strategies. Indeed, once fixed, it cannot be outperformed by any mutant strategy that might emerge. The study of stables through evolutionary game theory thus participates in the study of the problem of cooperation. By using analytical models studying populations as dynamic systems, the search for evolutionary stables is equivalent to the search for stable equilibria in the system. These classical approaches have made it possible to study the conditions necessary for the invasion of cooperators among a population of cheaters in different social dilemmas (Axelrod & Hamilton, 1981) or to study the balance of resource sharing in ultimatum games with partner choice (André & Baumard, 2011b).

1.3 Models of Partner Choice

Amongst all the mechanisms that allow the emergence of cooperative behaviour, partner choice has been studied by several models. The objective of these models is to propose different partner chice mechanisms and thus to identify the environments and behaviours adapted to the development of cooperation by partner choice. They also identify the constraints that need to be overcome in order for cooperation to evolve.

In this section, we present the key findings from the study of partner choice and the acquisition of cooperative behaviour through analytical and individual-centred models. These studies address one or more of the characteristics that must be present in the biological market for partner choice

[2]Tit-for-tat can be either Evolutionary Stable Strategy (ESS) or a Neutrally Stable Strategy (NSS) depending of the other strategies in competition. The difference between the two is detailed in Polak and Abdou (2014).

to be relevant. For example, the importance of market diversity is explored by McNamara et al. (2008). The study of the spatiality of the market is carried out by Aktipis (2011). The importance of partner choice strength is studied by Aktipis (2011), Barclay (2011), Campennì and Schino (2014), Debove, André, et al. (2015), McNamara et al. (2008). These studies show that implementations of partner choice can be cognitively very simple (Aktipis, 2011; Barclay, 2011; Debove, André, et al., 2015; McNamara et al., 2008) or require more complex structures such as memory (Campennì & Schino, 2014). Finally, constraints on market fluidity, i.e. the ease that individuals have to find a good partner and to keep it if they are satisfied, are examined by Campennì and Schino (2014), Debove, André, et al. (2015), McNamara et al. (2008).

1.3.1 Population Diversity

First of all, in order for an effective biological market and partner choice mechanism to exist, the population that constitute this market must be diverse in the quality of the individuals. Indeed, as stated in section 1.1.5 *Partner Choice and Biological Markets*, partner choice assumes that the recipient of an interaction chooses the best actor at its disposal as a partner. By having a choice between two actors with different qualities, the recipient of interaction will preferentially choose the actor who brings it the biggest gain. This preference of the recipients creates an arms race on the part of the actors. They must provide more payoff than the other actors in order to be chosen as partners.

If the population is too homogenous, that is, if all the players play identically, there is no point for the recipients to develop a trait of selectivity. The cost of developing and maintaining this trait would not be outweighed by the advantage of interacting only with efficient actors, since all of them are of equivalent quality. As the recipients do not possess a selectivity trait, which is a constituent part of partner choice, the actors are not in a competitive situation for cooperation. The cooperation arms race does not start.

McNamara et al. (2008) shows with an individual-based model the importance of the existence of behavioural variability in a population for effective partner choice. McNamara et al. (2008) constructs an aspatial model in which N individuals interact in random pairs in a continuous snowdrift game or a continuous Prisoner's Dilemma. Each individual has two characteristics: x, the level of investment of the individual in the interaction with the partner and y, the partner's investment acceptance threshold. All individuals are randomly arranged in pairs and play with their partners.

As a result of an interaction, individuals may decide to stay together or

leave, in a partner switching setup. If at least one of the individuals in the pair is not satisfied with his partner's investment, i.e. $x' < y$ or $x < y'$, then the interaction ends, and the two individuals will be randomly paired with the other individuals leaving an interaction for the next iteration. Each individual then reproduces in proportion to their payoff, and each descendant mutates according to a μ rate.

The results of these simulations are analysed by varying the rate of mutation μ and shows that the higher the mutation rate μ, the more individuals invest strongly and have a high acceptance threshold. This result comes from the fact that if the mutation rate μ is too low, then the population is too homogeneous. Thus, it is unlikely that leaving one's current partner will lead one to be with a better partner later on. This leads to not having an arms race in the selectivity threshold y . There is no interest for individuals to invest more in order to be accepted by their partner, and individuals act selfishly (cheat). If the mutation rate μ is large, then it is more likely that leaving a current individual will lead to a better partner, as it is more likely that a cooperative individual exists in a diverse population. It is therefore interesting to have a high selection threshold y.

1.3.2 The biological market in a spatialised environment

Partner choice models are mainly done in aspatial environments. In these environments, there is no notion of distance or proximity between individuals. Individuals are either randomly paired and separated to join a "pool" of single individuals, or they all interact with each other with diverses resource distribution systems. In a spatial environment, the search for a partner requires one to move in order to reach other individuals. Although models of partner choice in aspatial environments show that straightforward behavioural rules are sufficient to implement partner choice, it is tempting to think that in spatial environments, behavioural rules need to be much more complicated.

Aktipis (2011) shows that even in a spatial environment, it is possible to change cooperative behaviours through partner choice with elementary cognitive mechanisms. The behavioural rule that they call Walk Away allows the emergence of partner choice and could develop in many setups.

Aktipis (2011) proposes a model of partner choice in a spatial world constituted of a grid of cells, where individuals use their travel ability to choose the best group of partners. The model is similar to that of McNamara et al. (2008) but individuals are no longer paired randomly; they are paired based on their proximity. All individuals on a same cell play together. Once in-

dividuals have interacted, each individual has the option of staying or leaving (walking away) depending on the proportion of cooperators present in their cell compared to their satisfaction threshold. This satisfaction threshold is fixed for the whole population during the whole simulation, but the proportion of cooperators and cheaters varies according to an evolutionary algorithm.

The model shows that when the satisfaction threshold value is high, then the population stabilizes towards a predominance of cooperators. If the threshold value is low, then cooperators are exploited, and cheaters invade the population.

Aktipis (2011) thus presents an excessively simple behavioural rule in spatial environments that allows a partner choice in a population leading to the evolution of cooperation. The fact that individuals navigate in a complex environment does not necessarily imply that the cognitive mechanisms necessary for partner choice are very elaborate.

1.3.3 Competitive Helping

Partner choice leads to the existence of biological markets, in which individuals seek the best possible partner. As all agents share this objective, an individual must be the best possible to be chosen. Partner choice, therefore, leads to a "competitive helping" situation, an arms race. This market also leads to an assortative matching: The two best individuals cooperate together, the next two best individuals pair up together, and so on until the two worst partners pair up.

Barclay (2011) constructs a model testing the emergence of cooperative behaviour based on the strength of an experimentally imposed assortative matching. It shows the power of partner choice through what he calls "competitive helping". It is possible to make a population of individuals act cooperatively if each individual must, in order to obtain payoffs, be among the most cooperative so that other individuals in the population agree to give them the benefits of their work.

Thus, the purpose of this model is to study how individuals behave if they receive more or less reward from their investment compare to the ones of the others. Each individual a in the population A invests a value h_a, to form a total $H = \sum_{a \in A} h_a$. This total is then distributed among individuals according to their shares of investment. Assortative matching is represented in this analytical model by how agents share the resources produced. In order for assortative matching to be present, the agents investing the most must also be the agents that receive the most payoffs. In this model, the resource distribution is weighted by a z parameter that acts exponentially on

the distribution of resources amongst the individuals depending on their investment. If $z = 0$, then all individuals receive the same amount of resources regardless of their investment, creating no assortative matching. If $z = 1$, then the quantity of resources received for each individual is proportional to their investment, it is pure assortative matching. Finally, if $z =\rightarrow +\infty$, then the distribution of resources is a winner take all. In this case, the individual who invested the most gets all the payoffs. Barclay shows that the equilibrium investment h^* grows with z, that is, the more the shares of the payoff the agents get is correlated with their investment, the more they will invest in a social dilemma.

Barclay thus presents a model of "competitive helping". Individuals invest a lot to attract payoffs to them. In competitive helping situations, it is in the interest of individuals to invest as much as possible to obtain the output of others. The partner choice here greatly facilitates the emergence of cooperative behaviours.

1.3.4 Partner choice with memory

All the previous partner choice models presented here were based on partner switching, where individuals randomly find themselves with a partner and then decide whether to stay with it or not; or by a global allocation of resources according to their investment. These systems of partner choice require straightforward behavioural rules and do not require any memory on the part of individuals. Partner choice can also be driven with an upfront choice, thanks to the existence of memory in individuals. Campennì and Schino (2014) investigate partner choice based on memory.

The proposed evolutionary model is built with multi-agent simulations. The simulation is composed of N individuals. At each time step, an individual is designated as an actor and is proposed $n < N$ individuals. The individual a keeps in memory the last m interactions he had as a recipient. The individual chooses as recipient the individual who has had the most cooperative interactions with it in its memory repertoire. The actor can either cooperate (paying a cost of 1) to bring a gain to the recipient or cheat (paying nothing) and bring no gain to the recipient. The recipient records the outcome of the interaction in its memory. The behaviour of the individual is determined by its genome. The individual has a probability p of cooperating and $1 - p$ of cheating. After each episode, a new generation is formed by a proportional fitness reproduction and mutation of the probability p.

Campennì and Schino (2014) obtain as a result an assortative matching thanks to this partner choice mechanism, i.e. the individuals will cooperate proportionally more with the individuals who cooperate the most. This be-

haviour is possible when the memory size m of the individuals is large, and the number of individuals available n to choose is large. Moreover, the payoff of being a recipient of cooperation must be substantial, as is the quantity of cooperators present at the beginning of the simulation. Therefore, thanks to the implementation of memory in the agents as well as with behavioural rules that favour interactions with the partners who have cooperated the most, the emergence of cooperation by partner choice is possible.

1.3.5 Seeking Time and Interaction Time

In order for partner choice to be effective, the payoff obtained from cooperating with a successful partner must compensate for the loss of profit caused by the search for that partner. Indeed, it might be less advantageous to cooperate with no one for an extended period in searching for a good partner and then to cooperate with that partner for some time than to cooperate for a long time with a lousy partner. There is, therefore, a trade-off between the time spent searching for a good partner and the marginal gain from cooperating with that partner compared to the first partner encountered.

Debove, André, et al. (2015) construct an aspatial individual-based model with evolution to study the emergence of cooperation by partner choice and to show the importance of the cost of searching for a partner compared to the cost of choosing a bad partner.

In the model of Debove, André, et al. (2015), each individual has two traits, an investment trait $x \in [0, 1]$ and a selectivity trait y. At first, all individuals are in the pool of single individuals. At each time step, each individual has a β probability, defined by the model, of being paired with another individual to play an ultimatum game. The two individuals in the pair are randomly assigned as the proposer and the recipient. The proposer will make a proposition of sharing x of his payoff. The recipient refuses the interaction if $x < y$. In this case, both individuals return to the pool of single individuals. If $x \geq y$, the recipient accepts the interaction. The proposer receives for payoff $1 - x$ and the receiver x. The proposer and the recipient are then immobilized and no longer take part in any interaction. At each time step, they each have a τ probability of returning to the pool of single individuals. Thus, β is the probability of finding a partner, and τ is the probability of two individuals separating. The greater the β, the easier it is for an individual to find a partner. The smaller the τ, the more engaging the accepted interactions between two individuals over time.

Using simulations, mathematical analyses *and* experiments in humans, Debove, André, et al. (2015) shows that the greater β/τ is, the more the individuals act cooperatively (proposes a high and fair x, that is $x = 0.5$). If

β/τ is too low, then the cost of finding a better partner is too great to make refusing an interaction from a bad partner viable.

1.3.6 Discussion on partner choice modelling

All the studies presented above propose different systems in which cooperation evolves thanks to partner choice. This partner choice can be imposed experimentally (Aktipis, 2011; Barclay, 2011; Campennì & Schino, 2014) or it can also evolve (Debove, André, et al., 2015; McNamara et al., 2008). Different criteria necessary for partner choice are identified in these works. Effective partner choice needs to have a diverse population of cooperators but also cheaters so that the evolution of the partner choice behaviour provides a real evolutionary advantage (McNamara et al., 2008). Moreover, the partner search must be inexpensive, i.e., that it is easy to have access to many partners to sample before choosing the best one. This cost can be expressed in different ways. It can be a direct fitness cost, such as in the model of McNamara et al. (2008). Besides, this cost may be present in the access to information that individuals have, i.e. the number of individuals presented to them and the information they have about them (Campennì & Schino, 2014). Finally, the meeting rate with a potential partner is another expression of this cost. It is represented by the parameter β in the model of Debove, André, et al. (2015).

1.4 Evolutionary Robotics as an Individual-based modelling method for the evolution of cooperation

As seen previously, results presented before have been obtained with rather abstract models, especially when it comes to capturing the mechanistical constraints of the "real" world. Most models do not capture how the individuals actually move around, meet with potential partners and find resources. Even when they do consider spatial environment, such as Aktipis (2011), they do so in a much simplified form such by using grid-based 2D environment and fixed behavioural strategies.

In this thesis, we are interested in how Partner Choice benefits the evolution of cooperation *under more realistic constraints*. In particular, the probability of possible interactions (whether successful or not) depends on resources availability, population density and exploration strategies. Therefore, we design individual-based model that capture the complex interactions

at work in a pseudo-realistic 2-dimensional environment, where individuals learn to explore and interact, using realistic sensory inputs and actuators (i.e. continuous states and actions). This entails to also consider more complex decision-making apparatus, with two possible outcomes whether results obtained with more simplistic models may not hold anymore, or whether they do hold but must be further specified (e.g. by taking into account the fact the individual and resources are two different things).

In this Section, we present evolutionary robotics, which is the application of evolutionary computation to robotics. We present the various ways in can be used to enable swarm or collective robots to learn how to solve a task. We then present how the very same method can be, and has already been, used to tackle open questions in evolutionary biology, including the evolution of cooperation. When it comes to modelling for evolutionary biology, Evolutionary Robotics thus presents a ready-to-use method for individual-based modelling, where mechanistic contraints can be modelled as robotic agents move in pseudo-realistic 2-dimensional environment. This makes it possible to study how physical constraints imposed by the environment and the robotic agents may shape the evolutionary dynamics of learning to cooperate. And in our particular case, how partner choice may affect the evolution of cooperation in more complex setups.

1.4.1 Evolutionary Robotics and Collective Systems

Evolutionary Robotics uses evolutionary algorithms in the context of robotics. It can be described as a policy search method in the reinforcement learning framework. The goal is to optimize a set of control parameters, which are often weights of artificial neural networks, with respect to an objective function. It differs from other methods in reinforcement learning as it can deal with poorly informative objective function (delayed and sparse rewards) and continuous state and action spaces (Doncieux et al., 2015; Guide & S, 2003; Nolfi et al., 2000). It has been shown on several occasions to be competitive with other reinforcement learning methods (Salimans et al., 2017; Taylor et al., 2006; Verbancsics & Stanley, 2010).

There are many possible variations of an evolutionary algorithm. Figure 1.2 describes the general scheme: an initial population of N candidate solutions is randomly *initialized* and the *evaluated*. Solutions are then ranked according to their performance, and a *selection* process is applied to select a subset of the best solutions. Then, a new population of candidate solutions (e.g. particular control parameter values) is generated by applying *mutation* and *recombination* of the previously selected solutions. Then, the whole process starts again, until a termination criterion is reached, which can be

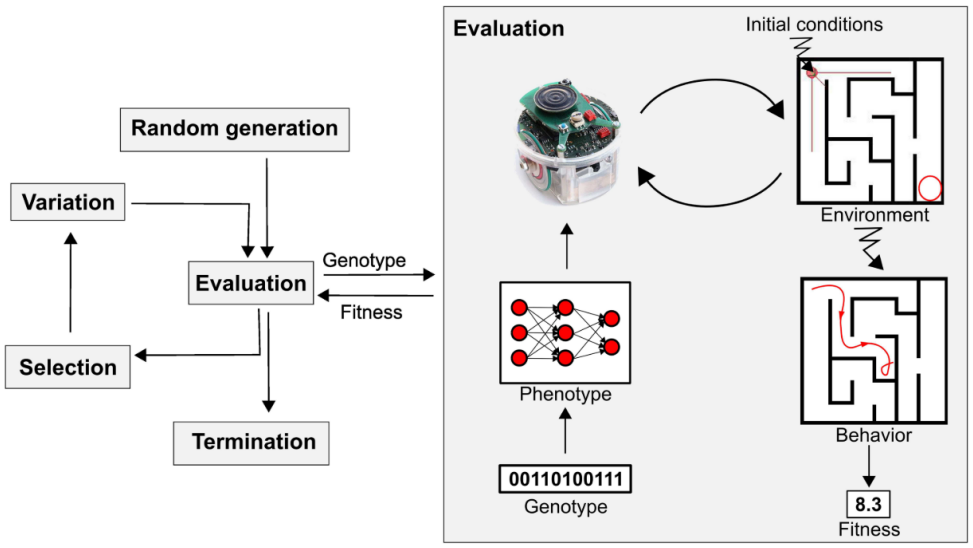

Figure 1.2: General scheme of an evolutionary algorithm. Taken from Doncieux et al. (2015). First, a random population of genotypes is generated. Each genotype is then evaluated. The genotypes are mapped into their phenotypes and the agents behave in their environment. Depending on their behaviours, each agent receives a fitness score. The evolutionary algorithm selects a subset of the genotypes according to these fitness scores, and create variation of these genotypes. These new genotypes are evaluated and the whole process starts again until a termination condition is met.

fixed beforehand depending on the amount of computational resource available, or dynamically triggered when the learning curve flatten or a particular performance value is attained.

In the context of collective and swarm robotics, the level of selection and team composition are to be considered during implementation. Waibel et al. (2009) proposed a classification along these two axes:

- Genetic team composition:
  - Homogeneous team: all robots are clones, they all make use of the same control parameter values, which implies that in similar conditions, they would act similarly. This can also be referred to as the *clonal* approach. ;
  - Heterogeneous team: all robots employ different control parameter values. One set of control parameter values thus codes for exactly one robot.

- Level of selection:
  - Individual selection: each robot is assigned its on performance assessment, which is usually a scalar value. Control parameter values of the robots with the highest scores are more likely to be selected to build a new population of candidate solutions, whether they were initially part of the same team, or not.
  - Team selection: performance is assessed for a whole team, with no distinction between participating robots, whether they are similar or not. All robots who were part of the teams with the highest are more likely to participate in the construction of a new population of candidate solutions, irrespective of their individual contribution to the team performance.

Several authors have explored the benefits and drawbacks of each combination of both axes. On the one hand, it was established that using (genetically) homogeneous team compositions provides scalability (Baray, 1997; Trianni & Dorigo, 2006), evaluation speed (Luke et al., 1998), and can demonstrate specialization (Bryant & Miikkulainen, 2003). On the other hand, using heterogeneous team composition provides more flexibility in terms of role specialization (e.g. specialists without environmental cues) (Baldassarre et al., 2003; Bernard et al., 2016; Bongard et al., 2000; Quinn et al., 2002).

More recent results refine these finding as Ferrante et al. (2015) showed that role specialization could be learned even within homogeneous teams as robots can dynamically switch roles depending on the environment. Conversely, Bernard et al. (2016) showed that in some extreme case requiring specialization, such as when leader/follower roles are required, heterogeneous team composition is mandatory to get a stable organisation between robots.

From a more general viewpoint, all these works in evolutionary collective robotics aim at obtaining a population of individuals that cooperate to maximize the sum of rewards at the population level. Combining homogeneous team with team selection is quite natural as all clones share the same goal modelled as a single objective function to optimize. Similarly, using an individual level of selection with heterogeneous teams is justified when individual performance is measured with respect to its own contribution to the global welfare of the whole population. This is an important assumption, which is difficult to hold in practice: the individual objective function must be crafted so that it should capture perfectly the contribution to the global welfare. An interesting side-effect is that using a heterogeneous team composition with a team-level selection has often been discarded as inefficient in evolutionary robotics by Waibel et al. (2009), even though this is at the core of many

works in the multi-agent domain, where proposing method to *estimate* individual contributions to the global welfare of the population has long been of paramount importance (Shoham, 2009; Wolpert & Nasagov, 2000).

For the sake of completeness, we will also discuss the particular case of distributed online evolutionary learning for robotics. In this setup, the lack of centralised control makes it mandatory to provide an individual objective function, ideally capable of evaluating the contribution of the focal robot to the global welfare (without having a direct access to the latter). Each robot acts as an individual learner, possibly exchanging information with neighbours. This is generally constrained by the robots' limited communication capability, which is typical of swarm robotics. Embodied Evolution (Bredeche et al., 2018; Watson et al., 2002; Watson et al., 1999) and Social Learning for Robotics (Heinerman et al., 2015) propose to use evolutionary operators in a distributed fashion to learn behavioural strategies with respect to an objective function which is expressed locally. In other words, each robot optimizes an embedded objective function, while the optimization is conducted in a distributed online fashion.

In distributed online evolutionary learning, the collective performance depends on the formulation of the local objective function, as in distributed welfare games (Marden & Wierman, 2008). The global performance can be achieved only if the Nash Equilibrium emerges naturally when each robot maximizes its *own* local fitness function. As stated earlier, the decomposition of a global objective into a local function is a challenging task and has been addressed in embodied evolutionary robotics from several perspectives. To date, this problem has been mostly studied for learning task specialization. Montanier et al. (2016) studied the necessary conditions for task specialization, which requires both geographical separation and sparse interactions, while Haasdijk et al. (2014) proposed to implement a market mechanism to counter-balance the convergence to a sub-optimal equilibrium.

1.4.2 Evolutionary robotics as a Method to Understand Cooperation in Nature

Understanding biological processes can be addressed in different and complementary manners, depending on the object of study, type of data, and the research question. Evolution, in particular, is difficult to observe in nature and our understanding of this process comes from a combination of field studies (Davies, 2012), experimental *in vitro* evolution (Kawecki et al., 2012; Lenski et al., 1998), mathematical modelling (Murray, 2002) and computational modelling (Railsback & Grimm, 2019; Schulze et al., 2017).

In the last 20 years, Evolutionary robotics has also been used as a computational modelling method to address research questions in evolutionary biology, extending the computational modelling toolbox. Compared to more traditional methods (e.g. (Nowak, 2006)), evolutionary robotics makes it possible to simulate the evolution of finite size population by controlling both the selection, replacement and variation processes. Various research questions have been explored so far, from the trade-off between efficiency and robustness in the evolution of communication (Wischmann et al., 2012) to the effect of relatedness in the evolution of cooperation (Waibel et al., 2011), from the evolution of cooperation with unrelated individuals (Bernard et al., 2016; Bernard et al., 2020), to the rarity of the evolution of reciprocity in nature (André & Nolfi, 2016).

In terms of method, evolutionary robotics for biology does not differ fundamentally from its use for engineering purpose. Differences are to be found in the subtle choices of operators used. In particular, fitness-proportionate selection (which simulates the Wright-Fisher model with selection, and is often discarded when optimal solution are to be engineered) is widely used when modelling evolutionary dynamics. However, evolutionary robotics for evolutionary biology also provides a unique feature as individuals can interact in a pseudo-realistic spatialised environment (Doncieux, 2015; Mitri et al., 2013; Trianni, 2014), emphasising the mechanistic aspect of interaction.

Waibel et al. (2011) illustrates a canonical example of the use of evolutionary robotics for the study of altruistic cooperation. The authors provide an experimental validation of Hamilton's rule, which state that one individual may choose to pay a cost to cooperate that depends on their level of relatedness of the recipient partner. More formally, Hamilton (1964) defined the condition to cooperate as $c < b \times r$ where c is the cost of the actor, b the benefit expected for the recipient, and r the relatedness between the individuals involved (cf. Section 1.1.3). In their model, Waibel et al. (2011) define a foraging task and observe the level of cooperation in different setups where the relatedness is artificially controlled by mixing clones and non-clones within the same 2-dimensional environment. Experimental results confirm the initial theoretical results *rigorously* but in a more realistic (though more complex) setup.

In a similar vein, Bernard et al. (2016) and André and Nolfi (2016) demonstrates that physical interactions can play an unexpected role in the evolution of cooperation. While cooperation can theoretically evolve, it often requires a very strong assumption about the ability for two (or more) individuals to learn *simultaneously* to physical coordinate, which is unlikely if individuals are not genetically related. Bernard et al. (2020) showed that alternative evolutionary pathways are possible in a more realistic setup, which could

not be captured in the initial abstract evolutionary game-theoretic setup: while it is unlikely to evolve the necessary traits for cooperation in several individuals at once, the problem of cooperation can be reformulated into a pure optimisation problem when individuals assume different roles. In this work, it is shown that collective hunting could arise from a leader-follower task decomposition: the leader benefits from an implicit cooperation from the follower and a single mutation in the leader can benefit both individuals (e.g. a larger prey that can only be hunted by two individuals becomes accessible with only the leader changing its strategy).

1.5 Thesis objective

The objective of this thesis is to use both individual-based models and environments from evolutionary robotics in order to identify which constraints can weigh on the evolution of cooperative behaviours with partner choice. The interest here is twofold: First of all, thanks to evolutionary robotics, we wish to identify better the mechanical and environmental constraints that weigh on the evolution of cooperation with partner choice, including in the living world. These constraints can arise from the complexity of genotypic-phenotypic mapping, but also, and above all, to problems of navigation and coordination between agents, resources access or population density. Reciprocally, we want to transfer the acquired knowledge on the emergence of cooperative behaviours in evolutionary biology in the field of evolutionary robotics in order to be able to design swarms of robots with distributed learning that could perform collective action efficiently despite individual learning.

In Chapter 2, we use tools from evolutionary robotics to design a computational model for studying the role of partner choice in the evolution of optimal cooperative behaviours among individuals. In particular, we study how resource *and* partner availability constrain the efficiency of partner choice in a population of individuals.

In Chapter 3, we extend our evolutionary robotics model to study how partner choice can improve the social behaviour of individual learners when resources are distributed in a spatial environment.

Beyond understanding the benefits and limits of partner choice in the evolution of cooperation, the results reported thereafter can also be used in a classic (off-line) evolutionary collective robotics approach. Even more, these results are also relevant in more general situations involving multiple *individual* learners, as the mechanism of partner choice works at the level of interacting robots.

In Chapter 4, we will actually explore the problem of partner choice and cooperation with individual learners. We will take a step back from biological considerations, and put the focus on the problem of learning cooperation with sparse rewards but without any assumption on the learning method used (e.g. deep reinforcement learning methods). Beyond evolutionary biology, the goal will be to address the problem of learning to cooperate in a multi-robots setup.

Chapter 2

Nothing better to do? Environment quality and the evolution of cooperation by partner choice

The work described in this Chapter has been submitted to Journal of Theoretical Biology and is currently under review. I am the first author and main contributor, in collaboration with my two supervisors J-B. André and N. Bredeche.

The effects of partner choice have been documented in a large number of biological systems such as sexual markets, interspecific mutualisms, or human cooperation. There are, however, a number of situations in which one would expect this mechanism to play a role, but where no such effect has ever been demonstrated. This is the case in particular in many intraspecific interactions, such as collective hunts, in non-human animals.

In this Chapter, we use individual-based simulations to propose a solution to this apparent paradox. We show that the conditions for partner choice to operate are in fact restrictive. They entail that individuals can compare social opportunities and choose the best. The challenge is that social opportunities are often rare because they necessitate the co-occurrence of (i) at least one available partner, and (ii) a resource to exploit together with this partner.

This has three consequences. First, partner choice cannot lead to the evolution of cooperation when resources are scarce, which explains that this mechanism could never be observed in many cases of intraspecific cooperation in animals. Second, partner choice can operate when partners constitute in themselves a resource, which is the case in sexual interactions and

interspecific mutualisms. Third, partner choice can lead to the evolution of cooperation when individuals live in a rich environment, and/or when they are highly efficient at extracting resources from their environment.

2.1 Introduction

Among the diversity of mechanisms put forward to explain the evolution of cooperation among non-kin, partner choice has been considered over the last twenty years as having probably played a particularly important role (Baumard et al., 2013; Bull & Rice, 1991; Eshel & Cavalli-Sforza, 1982; Noë & Hammerstein, 1994; Schino & Aureli, 2017; West et al., 2007b). When individuals can choose among several partners, which they can compare and compete against each other as in an economic market, this generates a selection pressure to cooperate more, in order to appear as a good partner, and attract others' cooperation (Noë & Hammerstein, 1994).

The effects of partner choice have been well described in many biological systems (Dittami, 2001). For example, in the interaction between cleaner fishes and their clients, the law of supply and demand determines how the added value of the interaction is shared, following market principles (Bshary & Grutter, 2006). When cleaners are rare, clients tolerate cheating on their part, while they become pickier when cleaners are numerous. The effects of partner choice have been documented in primate grooming, in meta-analyses showing that females groom preferentially those that groom them most and that a positive relationship exists between grooming and agonistic support (Schino, 2007; Schino & Aureli, 2008). In vervet monkeys, experiments have shown that individuals groom others in exchange for access to food, and do so for longer periods when fewer partners are available (Fruteau et al., 2009). The effects of partner choice have also been documented in humans, where it has been shown that the need to attract social partners is a major driver of cooperation (Barclay, 2016; Barclay & Willer, 2007; Baumard et al., 2013; Debove, André, et al., 2015; Schroeder et al., 2014). Besides, beyond cooperation partner choice also plays a decisive role in mating, leading to the evolution of secondary sexual characteristics, nuptial gifts, and assortative matching (Andersson & Simmons, 2006; Hammerstein & Noë, 2016; Zahavi, 1975).

On the other hand, there are many other biological situations in which one would typically expect partner choice to also play an important role, but where no such effect has ever been demonstrated. These include most intraspecific collective actions in non-human animals. The lack of partner choice is particularly salient in collective hunts such as colobus hunting in

chimpanzees, or pack hunting in carnivores. No empirical evidence in these species suggests that individuals cooperate for reasons related to partner choice, either to attract partners or to be accepted by them in their hunts. On the contrary, the majority of available data are consistent with the more parsimonious explanation that individuals are merely doing what is in their immediate best interest at any given time (Melis et al., 2008; Melis et al., 2011; C. Packer, 2014; G. Packer & Ruttan, 1988). In particular, if cooperation in collective hunts were driven in part by the need to appear as a good partner, individuals would be expected to willingly share the product of their hunts in a way that depends on everyone's actual engagement, to encourage participation in other hunts in the future. However, such voluntary and conditional sharing has never been documented in animal collective hunts (Melis et al., 2011). In evolutionary terms, therefore, collective hunting in these species is most likely an instance of *by-product* cooperation, rather than an instance of reciprocal cooperation based on partner choice. This lack of observation is all the more surprising given that, in similar collective actions, human behaviours are demonstrably driven by the need to appear as a good partner (Alvard & Nolin, 2002; Baumard et al., 2013). One may therefore wonder why the same effects did not produce the same consequences in other species.

Such a lack of observation could always be the consequence of the methodological difficulty in empirically proving the existence of partner choice, and more generally of conditional cooperation, outside humans (Henrich & McElreath, 2003; Raihani & Bshary, 2011). However, we would like to suggest an alternative here, namely that there is in fact a strong constraint impeding partner choice in many situations.

Partner choice requires that individuals can compare and choose among several opportunities for cooperation. In some cases, *partners* themselves are opportunities for cooperation and partner choice then only requires that partners are many and accessible. This is the case, for instance, in mating markets, or most instances of interspecific mutualism.

In other cases, however, finding an opportunity for cooperation requires more than just finding a partner. This is what happens when cooperation consists of several individuals working together to exploit environmental resources. In this case, a cooperation opportunity requires both a partner(s) and a resource, which imposes an additional constraint limiting the scope of partner choice. When resources are scarce, there are always few options to compare, and partner choice cannot operate. This could explain the lack of cooperation, beyond by-product cooperation, in many instances of collective actions in the wild despite the availability of potential partners.

To our knowledge, all models published so far on the evolution of coopera-

tion by partner choice focus on situations where finding a partner is sufficient to create an opportunity to cooperate. In this case, they show that partner choice can drive the evolution of cooperation in a relatively wide range of circumstances (Aktipis, 2004, 2011; André & Baumard, 2011a, 2011b; Barclay, 2011; Campennì & Schino, 2014; Debove, André, et al., 2015; Debove et al., 2017; Geoffroy et al., 2018; Johnstone & Bshary, 2008; McNamara et al., 2008; Noë & Hammerstein, 1994). Here, we wish to examine what happens on the contrary when resource availability constitutes a constraint on the operation of partner choice. To do so, we simulate the evolution of agents placed in an environment containing resources that can be exploited collectively. We show that, in a low-resource environment, and even if there are plenty of partners, partner choice is not able to drive the evolution of cooperation as individuals cannot pit the few cooperation opportunities against each other. What is more, we also show that the number of potential partners harms the evolution of cooperation when patches are scarce. When potential partners are numerous relative to the number of patches available, there are always too many individuals on any given resource as individuals have nothing else to do anyway. Hence, there is no point in trying to attract partners but on the contrary there are benefits in trying to limit their number. Partner choice is thus only effective when the number of available partners lies within a precise range of values, all the narrower as the availability of patches is low.

We believe that this constraint plays a central role in explaining that, in many species, although individuals do participate in collective actions, sometimes finely coordinating their behaviour with that of others, they do not actually seek to cooperate beyond what is in their immediate personal interest.

2.2 Methods

We build an individual-based model where we consider a population of N individuals living in an environment consisting of ω different patches on which resources are located. Each patch can host an unlimited number of individuals who gain payoff units by playing a modified version of the prisoners dilemma with all the other individuals present on the same patch (see details on the payoff function below). A simulation is composed of G generations, and each generation lasts T time steps during which individuals gather payoff units. At the end of the T time steps, individuals reproduce in proportion to their total payoff and die. During a time step, every individual is considered one by one in random order. When its turn comes, an individual evaluates

each of the ω patches of the environment, including the patch where it is currently located, assigns each a score (details in section 2.2.1), and then moves toward the patch with the highest score, or stays on its current patch if that is the one with the highest score. When an individual moves to a patch different from its current one, it incurs a cost c_m. Once every individual has taken its decision, individuals express their cooperation strategy on their local patch, and they collect a payoff that depends on their cooperation strategy, their partners' strategies, and the number of individuals present on the patch. Patches can disappear every time step, with a probability d, and are then immediately replaced by an empty patch. Table 2.1 sums up the parameters of the model. All data and source code used is available at https://osf.io/p5whz.

2.2.1 The decision-making mechanisms

The individuals' strategy in this environment consists of two separate decisions.

On the one hand, the individual must evaluate the different patches available and assign a score to each. This decision is made by an artificial neural network, called the "patch ranking" network. For each patch, this neural network has the following input information: (i) the number of other individuals already present on the patch, (ii) the average level of cooperation expressed by these individuals in the last time step, (iii) the level of cooperation that the focal individual would express should it join this patch, and (iv) a boolean that indicates whether the individual would have to move in space to join this patch (i.e. this boolean distinguishes the patch where the individual is currently located from all other patches).

On the other hand, the individual must decide on a level of cooperation once it is on a patch. This decision is made by another artificial neuron network called the "cooperation" network. As an input, this neural network only has the number of other individuals present on the same patch as the focal. We assume that the agent cannot modulate its cooperation level as a function of others' cooperation level. This assumption is meant to exclude the possibility that partner control strategies may evolve and allows us to focus only on the effect of partner choice (Noë & Hammerstein, 1994).

The details of the architecture of the neural networks are available in Section 2.5. The connection weights of both networks constitute the genome of each agent. They evolve by natural selection as exposed in section 2.2.4.

2.2.2 Phenotypic variability of cooperation

Each individual i present on a patch invests a given amount x_i into cooperation — where x_i is decided by the individual's cooperation network. However, as is now well established in the literature, selective pressures in favour of conditional cooperation stem from the presence of some variability in partners cooperative behaviour (see McNamara and Leimar, 2010). In order to capture the effect of variability in the simplest possible way, here we consider the effect of phenotypic variance in the expression of individuals' genes. At each generation of our simulations, each individual is subject to the effect of a *phenotypic noise* that modifies its cooperation level. If x_i^g is the cooperation level decided by the cooperation network of individual i, then the actual cooperation level played by the individual is $x_i = x_i^g + \epsilon$, where ϵ is drawn randomly as follows. The interval $[-1, 1]$ is uniformly split in N values, and every individual gets one value of ϵ chosen among these N values without replacement.

2.2.3 The payoff function

Individuals present on the same patch play a modified version of the n-player prisoner's dilemma. Consider a focal individual i playing x_i, in a patch on which there are $n-1$ other individuals whose average level of investment is $\bar{x}_{-i}$. The payoff of individual i is given by

$$P(x_i, \bar{x}_{-i}, n) = F(n) \times \left[ax_i + b\bar{x}_{-i} - \frac{1}{2}x_i^2\right] \tag{2.1}$$

where a represents the immediate, self-interested, benefit of the interaction, and b represents the social benefit of cooperation from others. When $a = 0$, the game is a prisoner's dilemma. In our simulations, however, we always choose $a > 0$ which entails that cooperation always has a slight immediate benefit. This assumption is necessary to avoid a bootstrapping problem in the joint evolution of cooperation and partner choice, which is an important issue but not the object of the present study (see André, 2014 for a similar issue in the case of partner control).

The function $F(n)$ is meant to capture the fact that there is an optimal number of individuals exploiting a patch and is given by

$$F(n) = e^{-(n-\hat{n})^2/(2\sigma^2)} \tag{2.2}$$

where $\hat{n}$ is the optimal number of individuals per patch and σ measures the tolerance to variations in the number of individuals per patch (i.e., σ^{-1} measures the strength of the penalty that stems from being a suboptimal number

of individuals on the same patch). When tolerance σ is very low, individuals get a benefit almost only when they are exactly the optimal number $\hat{n}$ on a patch. On the other hand, when tolerance is very large, there is almost no penalty for being too many, or too few, partners per patch.

With this payoff function, in the absence of partner choice, the evolutionarily stable strategy is to invest $x_{ESS} = a$, whereas the "socially optimal" cooperation level, that is the level that would maximise the average payoff of individuals on the patch, is $\hat{x} = a + b$.

2.2.4 The evolutionary algorithm

Each individual has a genome composed of the weights of its two neural networks, which makes a total of 84 genes $g = (g_1, \ldots, g_{84})$ with $g_i \in]-10, 10[$. We consider a population of fixed size M. The first generation is composed of M individuals with random genes for the neural network weights, drawn uniformly in $]-1, 1[$. We then use a fitness proportionate evolutionary algorithm to simulate evolution. After the T time steps of a generation have taken place, individuals all reproduce and die. A new population of M individuals is built out of the previous generation by sampling randomly among the M parents in proportion to their cumulated payoff, according to a Wright-Fisher process.

A mutation operator is applied to each offspring. Every gene of every offspring has a probability μ to mutate and a probability $1 - \mu$ to stay unchanged. If a gene g_i, with value v_i, mutates, it has a probability 0.9 to mutate according to a normal distribution and thus reach a new value sampled in $\mathcal{N}(v_i, 0.1)$ and a probability 0.1 to mutate according to a uniform distribution and thus reach a new value sampled in $\mathcal{U}(]-10, 10[)$.

The evolutionary algorithm is run for G generations.

In our analyses, we will vary N, which represents the number of individuals present together in the environment (i.e. the social population size). However, we want to keep constant the demographic population size ($M \geq N$) so as not to alter the relative strength of drift and selection. To do so, we create $\lceil N/M \rceil$ parallel environments. The M individuals of the demographic population are then randomly assigned so that each environment has exactly N individuals. For the last environment to be completed, randomly chosen genetic individuals are duplicated, but their payoff in this environment is then not considered for the calculation of their fitnesses.

Parameter	Description	Value
Environment		
M	Demographic population size	100
d	Probability of the disappearance of patches, per time step	1/1 000
T	Number of time steps per generation	1 000
c_m	Cost of moving to another patch	0
N	Social population size	variable
Payoff		
a	Immediate personal benefit of cooperation	5
b	Social benefit of cooperation	5
$\hat{n}$	Optimal number of individuals per patch	variable
σ	Tolerance to variations in the number of individuals per patch	variable
Evolution		
G	Number of generations	1 500
μ	Probability of mutation per gene per generation	0.01

Table 2.1: Parameters of the simulation

2.3 Results

2.3.1 Cooperation cannot evolve when patches are scarce

We simulate the evolution of a population of $N = 100$ individuals for $G = 1500$ generations, for different values of the number of resource patches ω, but always in a situation where the optimal number of individuals per patch was $\hat{n} = 2$. Cooperation only evolved when patches were more abundant than a threshold (Fig. 2.1a). This can be understood as follows. When resource patches are few, precisely when $\omega < \frac{N}{\hat{n}}$, individuals have little cooperation opportunities and there are therefore always more individuals per patch than what would be optimal (in this case, the optimal number of individuals per patch is $\hat{n} = 2$). As a result, additional individuals joining a patch are more of a nuisance than a benefit, and there is therefore no benefit in trying to attract partners by appearing cooperative. On the contrary, when the number of available patches is non-limiting, individuals have many social opportunities and it is therefore worth investing in cooperation to attract partners. This second situation also captures what happens when individuals themselves constitute a resource.

We simulate the evolution of cooperation in situations where the optimal number of individuals per patch, $\hat{n}$, is larger (Figs. 2.1b, and 2.1c). Overall, the outcome is even less favourable to cooperation. This may seem paradoxical but can be understood as a consequence of the law of large numbers. When the number of individuals per patch is large, whether it is greater or less than $\hat{n}$, the effect of each individual on the average quality of its patch is very small anyway. There is therefore little value for an individual to invest in cooperation to try and attract partners.

We also run simulations where we vary the tolerance coefficient σ and find that the higher the tolerance on the number of individuals on a patch (i.e. the higher σ), the less cooperation is favoured by evolution (results are shown in Fig. 2.1d in the extreme case where the number of individuals per patch has no impact at all on the payoff, and for the general case in Fig. 2.4 of Section 2.5). This result can also be understood because there cannot be any benefit in attracting partners when the number of individuals per patch does not matter.

Lastly, we also vary the cost of moving to another patch, c_m , and find that the higher the cost, the less cooperation is favoured, as expected from the literature on partner choice (Section 2.5, Fig. 2.5).

Overall, the evolution of cooperation by partner choice can only take place in the restricted conditions where (i) there is an optimal number of individuals per resource patch, (ii) this optimal number is low, and (iii) the

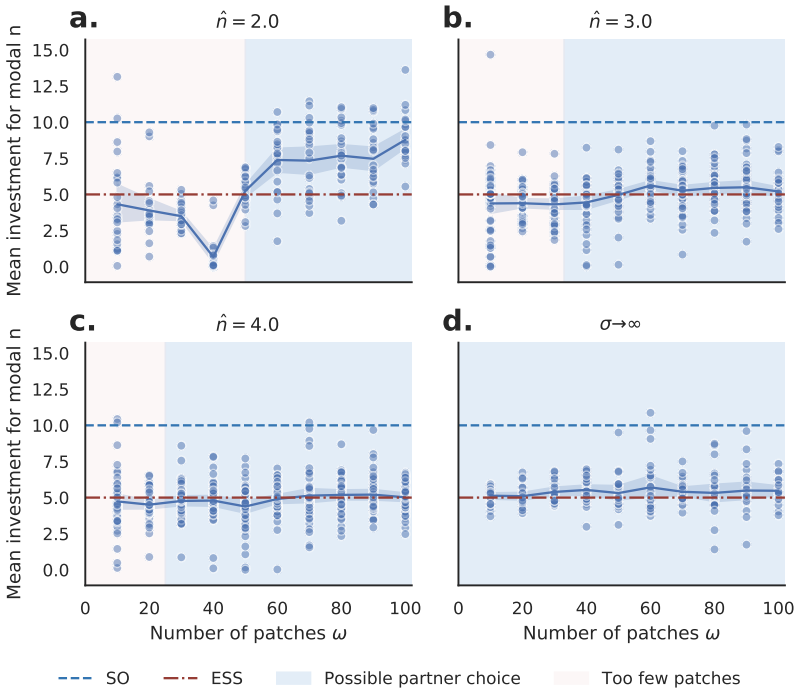

Figure 2.1: Mean investment in simulation for different numbers of opportunities ω and a fixed population of $N = 100$ individuals. Results after 1 500 generations. **a.** When $\hat{n} = 2, \sigma = 1$ Cooperation evolves when $\omega \geq 50$. **b-c.** For $\hat{n} \geq 3, \sigma = 1$, cooperative behaviours never evolve. **d.** When $\sigma \to \infty$, there is no pressure for individuals to attract partners and cooperative behaviours never evolve.

number of resource patches in the environment is large.

2.3.2 Cooperation cannot evolve when there are too many partners around

In a second step, we simulate again the evolution of a population of $N = 100$ individuals for $G = 1500$ generations in a situation where the optimal number of individuals per patch is $\hat{n} = 2$, but this time we hold the number of patches constant, $\omega = 20$, while varying the actual number of individuals, N, present together in the environment.

In this case, cooperation only evolves when the number of individuals in the environment is intermediate. This can be understood as follows. When the number of individuals in the environment, N, is too close to the number of individuals $\hat{n}$ that are needed to exploit at least one patch — or even more

so when $N < \hat{n}$, then the number of available partners is limiting. As a result, the actual number of cooperation opportunities from which individuals can choose is very low, partner choice is thus a weak force, and the benefit of investing into cooperation is low. On the other hand, when the number of individuals in the environment N is larger than the total number of individuals that can be accommodated on the available patches, that is when $N > \hat{n}\omega$, the number of available patches is limiting. In this case, we find the result described above (Fig. 2.2a). The problem is rather that there are always too many individuals on each patch than too few and partner choice is also a weak force. There is, therefore, a range of intermediate population densities, neither too low nor too high, for which cooperation can evolve.

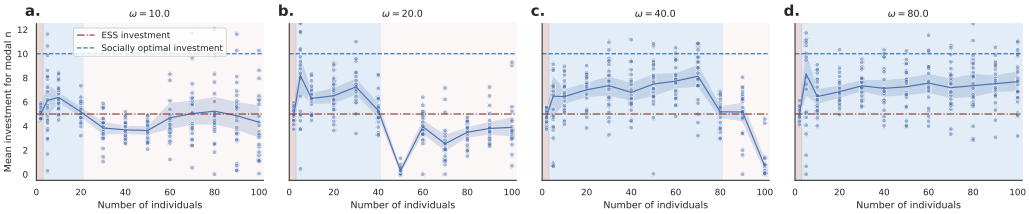

Figure 2.2: Effect on the population size in the environment with 20, 40 or 80 patches and an optimal number of agents $\hat{n} = 2$ and $\sigma = 1$. Agents have a cooperative behaviour for $\hat{n} < N < \omega \times \hat{n}$.

We then perform the same simulations again, but with more patches available in the environment (i.e. for larger ω, Figs. 2.2b and 2.2c). We observe that the range of population densities for which cooperation could evolve was then broader. This can again be understood in the above framework. On the one hand, the lower boundary of population density, $N \approx \hat{n}$, below which the number of individuals is a limiting factor, is unaffected by the number of patches available. On the other hand, the upper boundary of population density, $N > \hat{n}\omega$, above which the number of patches is a limiting factor, increases with the number of patches, ω. As a result, the width of the range of population densities where partner choice is effective increases.

2.3.3 Analysis of the behaviour of "patch ranking" networks

We analyse the response of the patch ranking networks that were present in our simulations after 1500 generations. To do so, we feed the patch ranking network of each agent with fake patch values (varying the partners' investment, the focal individual's investment and the number of individuals present

on the patch) and evaluate their response. We adjust the score of each patch to allow for comparisons between individuals and simulations (see details in Section 2.5).

First, we analyse the networks that evolve when the number of patches in the environment is non-limiting such that cooperation and partner choice evolve (Figs. 2.3a and 2.3b). In this case, we find that the patch ranking networks always prefer to move to patches (i) where there is exactly one partner already present (Fig. 2.3a) and (ii) where the partner's level of cooperation is the highest (Fig. 2.3b), which confirms that partner choice, and more generally patch choice, is at work in this condition.

Second, we analyse the networks that evolve when the number of patches in the environment is highly limiting such that cooperation and partner choice cannot evolve (Figs. 2.3c and 2.3d). In this case, we find that the patch ranking networks always prefer to move to patches (i) where there are already two or three partners present, (Fig. 2.3c) and (ii) where the partners' level of cooperation is close to the selfish optimum level of cooperation (Fig. 2.3d). This confirms that, in this case, partner choice is not a driving force able to lead to the rise of cooperation.

2.4 Discussion

Partner choice can lead to the evolution of cooperation when individuals can compare several opportunities for social interaction and choose the most advantageous ones. In this Chapter, we have shown that the conditions for this to happen are, however, quite restrictive. They entail that individuals truly have access to a range of social opportunities. However, in many cases, social opportunities are rare because they necessitate the co-occurrence of two things at the same time: (i) at least one available partner, and (ii) an exploitable resource or, more generally, "something to do" with that partner. In this Chapter, we have used individual-centred simulations to study the consequences of this constraint on the evolution of cooperation by partner choice. We have obtained the following results.

First, partner choice cannot lead to the evolution of cooperation when resources are scarce, and therefore opportunities for cooperation are rare. This explains why, in many species, social interactions show no evidence of cooperation beyond immediate self-interest (Bullinger et al., 2011; Melis et al., 2011; Scheel & Packer, 1991). Even when individuals engage in collective actions, for example when they hunt collectively, others have so few alternative opportunities anyway that there is no need to seek to draw them into the collective actions. They will come anyway, for want of anything better

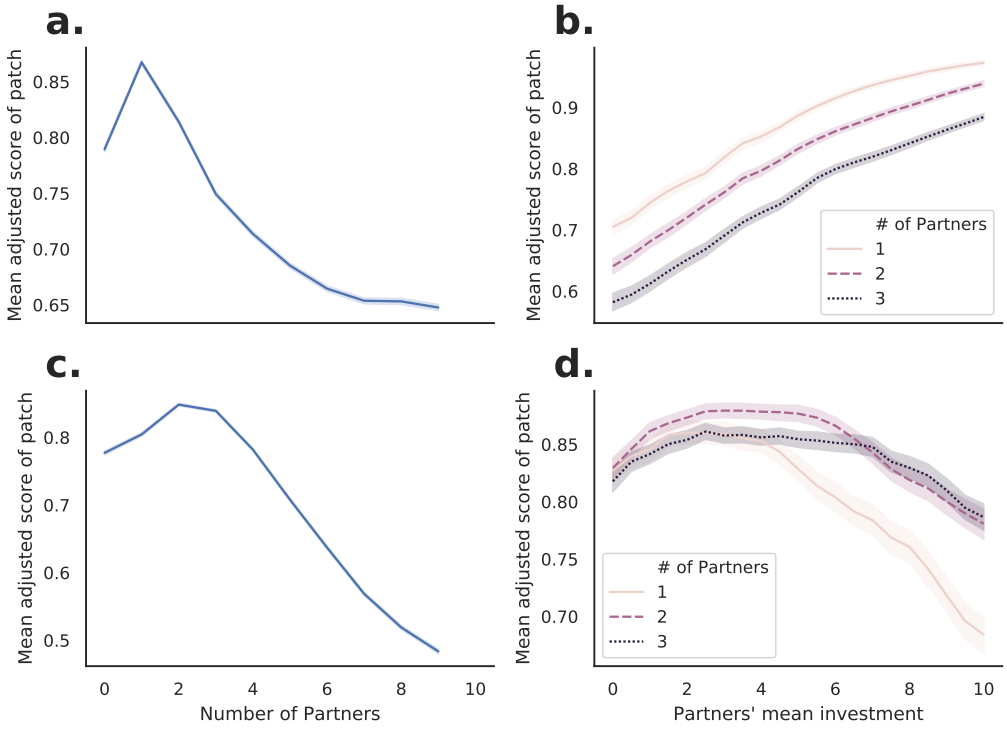

Figure 2.3: Mean score of patches as evaluated by the patch ranking network of 100 individuals in 24 simulations, with $N = 100$, $\hat{n} = 2$, and $\sigma = 1$. The investment of the focal agent is set to the value it would have invested in the context of the evaluated patch. **a**. Mean patch score as a function of the number of partners already present with $\omega = 80$. Individuals have a clear preference for patches with a partner already present (therefore with two individual including themselves) **b**. Mean patch score as a function of partners' mean investment with $\omega = 80$. Individuals always prefer the most cooperative partner available. This is characteristic of a partner choice response. **c**. Mean patch score as a function of the number of partners already present with $\omega = 20$. Individuals prefer patches where there are already two or three individuals on the patch, even though the optimal number of individuals on a patch is $\hat{n} = 2$. **d**. Mean patch rank as a function of partners' mean investment with $\omega = 20$. Individuals prefer patches where agent invests around the value $x_{ESS} = 5$ or do not have a clear preference about their partners investment. Partner choice mechanism has not developed.

to do. Even worse than that, as opportunities for cooperation are rare, not only are there always enough partners in each collective action without it being necessary to actively attract them. In fact the opposite is true: There are always too *many* individuals participating in each cooperation endeavour (see Fig. 2.2). This has been documented for instance in pack hunting in Lions, where Packer showed that lionesses often hunt in groups that are too large compared to what would be optimal (C. Packer et al., 1990). In such a case, the average gain per individual in a collective action is reduced and not increased by the participation of others, and there is therefore no selection to attract partners but rather a selection to push them away at the time of sharing.

Second, partner choice can lead to the evolution of cooperation when partners constitute in themselves resources. There is, in this case, no further requirement for a social opportunity than the need to find a partner. This occurs, for instance, in sexual markets, or in the many instances of interspecific mutualisms, where the other individual alone constitutes an opportunity to cooperate. It is therefore understandable that partner choice plays a particularly important role in these two types of interactions (Andersson & Simmons, 2006; Bshary & Grutter, 2002; Schino & Aureli, 2008).

Third, partner choice can lead to the evolution of cooperation when the environment is rich or, said differently, when individuals are efficient at finding opportunities for cooperation in their environment. Living in an environment rich in opportunities, and/or having skills that increase the effective number of opportunities one can exploit, brings with it the possibility of *choosing* between different opportunities. This puts greater pressure on individuals, who are then competing to attract partners on their own opportunity, rather than on another, and thus selects for cooperation beyond immediate self-interest.

This result could suggest a potential relationship between the evolution of cooperation on the one hand, and the evolution of cognitive abilities to more efficiently extract resources from the environment on the other. The possibility that there is an evolutionary relationship between cooperation, or more generally sociality, and cognitive capacities has long been discussed in the literature, and several hypotheses have been proposed to explain it (e.g. (dos Santos & West, 2018; Dunbar & Shultz, 2007)). These hypotheses, however, are all about the joint evolution of sociality with cognitive capacities that are *specifically* dedicated to social life itself. The present results suggest that cognitive abilities that have nothing to do with cooperation or sociality per se, namely the sheer ability to extract resources from the environment, could also play a role in the evolution of cooperation. This occurs because enhanced cognitive abilities allow transforming and extracting high-

value resources from the environment (Kaplan et al., 2000), thereby creating more opportunities for cooperation. As a result, a given environment contains more opportunities for cooperation for individuals with strong cognitive skills, such as human beings, than for the individuals of other species. This then affects the state of the market for cooperation, increasing the amount of competition between alternative social opportunities, thereby selecting for more investment into cooperation to attract partners.

2.5 Supplementary Materials

Architecture of the Artificial Neural Networks

The decision module of each individual is composed of two artificial neural networks: The "patch ranking" network and the "cooperation" network.

The "patch ranking" network is a multilayer perceptron with six inputs and one bias, one hidden layer composed of ten neurons and one bias, and one output. The activation function of the network is tanh. The inputs of the neural network are the (i) the investment value of the individual if it comes to the patch, (ii) the mean investment value of its partners if it comes to the patch, the number of partners on the patch split in (ii) units, (iv) tens and (v) hundreds and (vi) the cost of moving to the patch. All inputs are normalised between zero and one. The number of partners is split into units, tens and hundreds to allow better discrimination of small variations in the number of partners on a patch when there are a too large number of individuals in the environment (N is large). In total, the number of weights in this neural network is 73.

The "cooperation" network is a multilayer perceptron with two inputs and one bias, one hidden layer composed of three neurons and one bias, and one output. The activation function of the network is tanh. The inputs of the network are the number of partner on the patch split (i) in units and (ii) tens. In total, the number of weights in this neural network is 13.

For both networks, all weights are initialised in the range $[-1, 1]$ and their boundaries are $[-10, 10]$.

Adjusted score computation

To study the response of the patch ranking network between several individuals and simulations, we cannot use the raw output of the network of each individual. Indeed, what matters for the effectiveness of the network is the *relative score* between patches for *one individual*. Therefore, two individuals

can have the same patch preference ranking even if they do not use the same scores for each patch.

To take this issue into account, for each individual, we compute the statistical rank of all the 210 fake patches tested according to the scores of the patch ranking network. That is, the patch with the lowest score get the value 1 and the patch with the highest score get the value 210. The statistical rank is computed using the "max" method. In the case of equality — which often happens as the network saturates in its boundary values, the rank given to all the equal patches is the highest rank.

Then, we normalise the ranks for legibility. We divide the rank of each patch for each individual by the highest rank given (210) to get a score between 0 and 1. Finally, we average the score a patch between all individuals and simulations for a given condition, giving us its average normalised score for this condition.

Variation on the tolerance strength

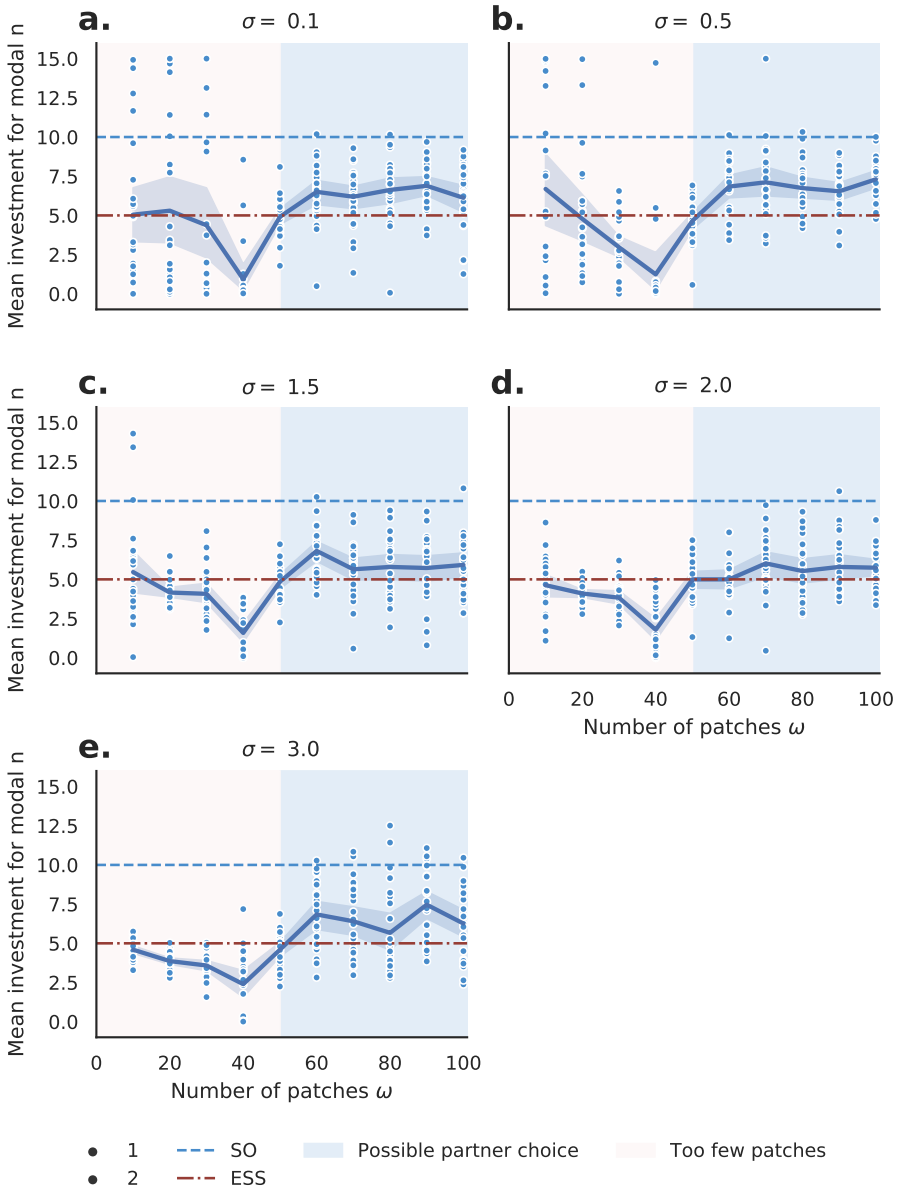

Figure 2.4: Mean investment in simulations for different numbers of opportunities ω, different values of tolerance strengths σ and a fixed population of $N = 100$ individuals. Results after 1500 generations. **a-b.** When the tolerance strength is strong (i.e. $\sigma \leq 1$, see Fig. 2.1a for $\sigma = 1$), agents cooperate. **d-g**. When the tolerance strength is low (i.e. $\sigma \geq 1.5$), agents do not cooperate. This is explained by the fact that too many agents (including cheaters) can come on the resource without suffering a tolerance that has a strong impact on the gains. So there is a dilution effect of responsibility that sets up in the same way as when $\hat{n}$ is big.

Effect of the cost of moving

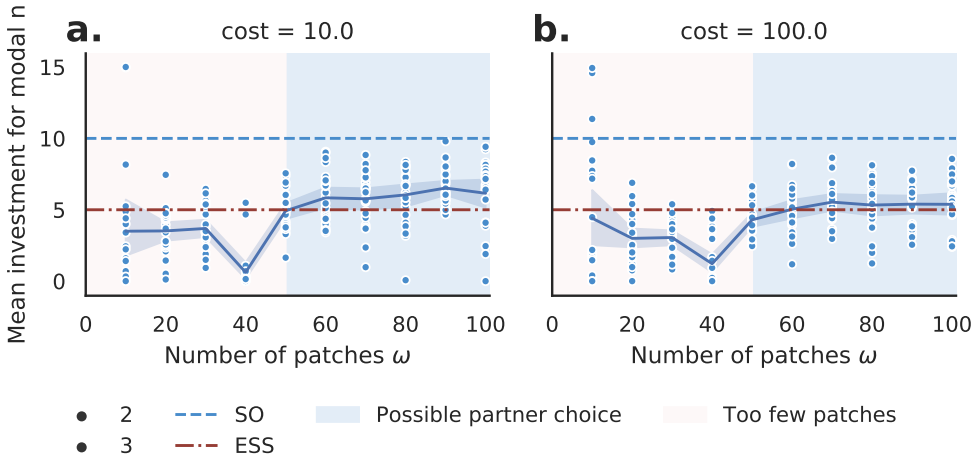

Figure 2.5: Mean investment in simulations for different numbers of opportunities ω, different values of the cost of moving and a fixed population of $N = 100$ individuals. Results after 1500 generations. The reference figure when the cost c_m is 0 is available in Fig. 2.1a. The greater the cost is, the less cooperative the population is. Increasing the cost of moving increases the cost of partner choice. When the cost is too high, it is of no interest for the agents to cooperate to attract new partners, as if a cheater joins them, it will be too costly for them to leave the opportunity with a defector.

Chapter 3

Learning to Cooperate in a Socially Optimal Way in Swarm Robotics

The work described in this Chapter has been published in the proceedings of the Conference on Artificial Life (ALIFE-2020), of which I am the first author and main contributor, in collaboration with my two supervisors N. Bredeche and J-B. André.

In this Chapter, we question the limits of computational models that result from using non-spatialized models (as used in the previous Chapter), *i.e.* when agents do *not* have to navigate in their environment to find a partner or a resource. With non-spatialized models, the environment considered is not ecologically realistic as its dynamics depends from control parameters of the model, provided beforehand by the experimenter. Furthermore, these models are simple, and evolution is rather straightforward as each gene binds to one and only specific function. The question remains open as to whether the results obtained earlier still hold in a more complex environment.

Throughout this Chapter, we study how the explicit consideration of space affects the evolutionary dynamics of cooperation in a particular context, that of evolutionary swarm robotics. By using a swarm robotics setup, we expect to gain a two-fold contribution. First, swarm robotics simulations offer a natural extension towards spatialized environments with respect to the model described in the previous Chapter. Secondly, partner choice may provide an efficient mechanism to be used within a collection of learning robots, whenever each robot is considered as independent from the others with respect to its own objective function. While the evolutionary algorithm used here works in a centralised fashion, we assume that individuals are in com-

petition with one another, so that the expected best strategy is to maximise each individual's gain, which may (or may not, depending on the context) imply cooperating with others.

In the case of a pseudo-realistic environment with navigation control, a simple genotype-phenotype mapping is no longer possible. There is no simple way to design an efficient navigation algorithm adapted to the environment. Individuals must navigate in the most efficient way to find partners, but the most efficient navigation rules depend on individuals and resources density, as well as on other individuals' behaviours. Furthermore, the search time can no longer be a simple parameter given by the experimenter, as it will depend on the individuals' ability to navigate and to find each other.

We show that results on partner choice still hold when a more realistic environment is considered. Evolving cooperation with partner choice proves surprisingly efficient in our swarm robotics setup, when the swarm size is large, and the duration of interaction is long. This confirms and extends results obtained in the previous Chapter, but it also paves the way for implementing partner choice as a mechanism for learning how to cooperate in a collective of independent robotic agents, whether learning is accomplished by an evolutionary algorithm or any other *single-agent* reinforcement learning algorithm (i.e. each robotic agent embeds its own learning algorithm[1]).

3.1 Introduction

Nature abounds with impressive examples of swarm intelligence (Camazine et al., 2001), which have motivated researchers in robotics to devise methods to obtain self-organizing behaviors in collective of robots (Bayindir, 2016; Beni, 2005; Brambilla et al., 2013; Hamann, 2018; Mataric, 1992).

However, designing the behaviors of a robot swarm poses a challenge in itself as collective behaviors emerge from the multitude of interactions between robots, and are therefore difficult to predict and design. The use of automatic design methods can circumvent this problem to some extent, but is based on constraining assumptions as they generally assume that (1) the collective payoff is known and available and (2) the learning algorithm can be iterated in a centralized way.

This is the case in multi-agent reinforcement learning methods for decision making under uncertainty (Amato et al., 2015) and in evolutionary swarm robotics (Trianni, 2008a). In the latter, these two assumptions enable to use homogeneous populations, ie. swarms of clones (Hauert et al., 2019; Trianni, 2008b). Learning with clonal populations have been shown to provide several

[1]The next Chapter explores this direction.

advantages: it can lead to purely altruistic behaviors (Waibel et al., 2011), it can deal well with credit assignment (Waibel et al., 2009), and it allows the acquisition of specialized behaviors even though all robots share the same control parameters (Ferrante et al., 2015; Tuci & Trianni, 2014).

In this Chapter, we lift the two previously mentioned hypotheses. We are interested in a population where all individuals are different and get individual payoffs (without knowing about the global payoff). While we use a classic evolutionary algorithm scheme, this setup is relevant for two other learning settings: individual learning facing collective tasks (Fudenberg, 1998) and distributed on-line reinforcement learning (Bredeche et al., 2018; Heinerman et al., 2015). In the class of problems addressed by either methods, cooperation is possible only when the individual's objective is aligned with the global objective, which requires a carefully designed individual objective function.

As this may not always be the case, we address the following question in this Chapter: **how to enable each robot in a swarm to learn the *socially* optimal behavior when this behavior is *individually* sub-optimal?**

This problem has been extensively studied in game theory and evolutionary biology (Axelrod & Hamilton, 1981). Whenever the accomplishment of a task by a group of individuals is *not* aligned with each individual's objective, maximizing one's own payoff will interfere with the execution of the collective task unless explicitly constrained otherwise.

Several mechanisms have been identified that allow the alignment between the individual's and the global objectives (West et al., 2007a). Among these mechanisms, partner choice is revealed to be particularly efficient. If all individuals can choose with whom to cooperate, then it is in everyone's interest not only to choose the best partner, but also to cooperate so as to be chosen. In other words, there is a selection pressure that favours those individuals who are able to make a good compromise between self-interest and common interest.

Earlier results obtained in theoretical biology have shown that for partner choice to be effective, the time spent searching for a partner compared to the time spent interacting with partners should be as short as possible (Debove, André, et al., 2015). However, all studies so far consider tightly controlled conditions, either with learning partner choice occurring in a well-mixed population (McNamara et al., 2008) or with the agents moving in a discrete grid world but without learning (Aktipis, 2011).

We propose an implementation of the mechanism of partner choice for evolutionary swarm robotics, the use of which is learned by the robots depending on the interactions between robots and the task to be performed. We also propose a study of the necessary conditions for partner choice to

enable the learning of a socially optimal cooperative strategy when such a strategy is not naturally stable (i.e. not a Nash Equilibrium). In agreement with theoretical results from evolutionary biology, we show that the use of partner choice in a robot swarm can shift the Nash Equilibrium from using a sub-optimal defective strategy to using a cooperative strategy. However, we also show that severe constraints over the number of encounters and the duration of interactions are key order parameters to enable the learning of socially optimal cooperative strategies.

This Chapter is structured as follow. Firstly, we present a foraging task where robots must pair to harvest resources spread in the environment. This task is defined so that selfish individuals are favored by selection pressure, even though this is detrimental with respect to maximizing the payoff, as in a typical social dilemma. Secondly, we describe the implementation of the partner choice mechanism and the details of the learning algorithm and controller representation used. Thirdly, experiments are conducted to assert the relevance of partner choice, and its limits, in enabling the learning of socially optimal strategies. Several control experiments are also presented to further validate our results with respects to possible open questions in the model.

3.2 Methods

3.2.1 Environment

We define a collective foraging task where N robots move and consume resources in a circular arena (see Fig. 3.1). The environment in which the robots move is continuous. The robots are subject to a simple kinematic model, and can control their translation speed and angular speed. Resources are small objects spread randomly throughout the arena, and can be consumed *only* if two robots are into contact with the resource at the same moment. Once a resource is consumed, it disappears and a new resource appears at a random location in the environment to ensure that the density of resources in the environment remains constant over time.

In order to consume a resource, both robots in a pair must invest some amount of energy, which is learned and may differ from one robot to another (see Section Learning). Each robot receives a payoff based on its own investment and that of its partner, for each resource harvested. This means that each robot has to make a compromise between the effort made to harvest the resource and the expected payoff.

We define the experimental setup so that a robot may either cheat (mini-

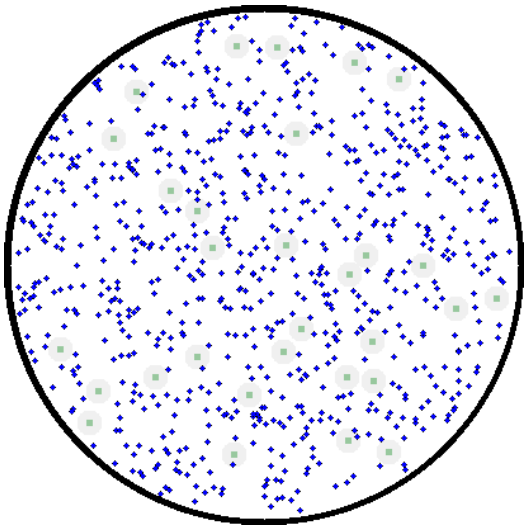

Figure 3.1: The environment is a circle arena. Blue dots are robots. Green dots are resources. Robots can see the resources, and when two robots are close enough (light grey area), they may interact together to forage the resource. The Roborobo simulator is used (Bredeche et al., 2013).

mizing its own investment while maximizing its gain) or cooperate (maximizing the gain of the pair). This is achieved by defining a payoff function such that the Nash Equilibrium corresponds to a selfish behavior, while the social optimum where robots cooperate is not stable (see Section Payoff function).

3.2.2 Payoff function

When two robots interact with each other, they earn a gain determined by the investment of the two agents. The gain of an agent a_i investing x_i with its partner a_j investing x_j is determined by the function $P(x_i, x_j)$ described in the equation 3.4.

$$PG(x_i, x_j) = \frac{a}{2}(x_i + x_j) \tag{3.1}$$

$$PD(x_j) = \frac{b}{2}(x_j) \tag{3.2}$$

$$C(x_i) = \frac{1}{2}x_i^2 \tag{3.3}$$

$$P(x_i, x_j) = PG(x_i, x_j) + PD(x_j) - C(x_i) \tag{3.4}$$

This function is a mixture of a public good (PG, modulated by a) and a

prisoner's dilemma (PD, modulated by b) and a quadratic cost C. For a_i to maximize its individual gain ($P(x_i, x_j)$), the optimal investment is $x_d = \frac{a}{2}$, which corresponds to the defective behavior. For the group to maximize their total gain, both agents must invest $\hat{x} = a + \frac{b}{2}$, which corresponds to the cooperative behavior. By using a combination of two classical social dilemma games, we ensure that (i) the optimal selfish individual investment x_d is greater than zero (which remove possible boundary effects) and (ii) there is a clear-cut difference between cooperative and selfish strategies.

Figure 3.2 plots the payoff function with different partner's investment values. The maximum payoff for the focal robot is always obtained for $x_d = \frac{a}{2}$, notwithstanding the contribution of the partner. However, if both agents play the same strategy, the maximum payoff for the group is found at $\hat{x} = a + \frac{b}{2}$. We define x_d as the contribution corresponding to a defective strategy, and $\hat{x}$ as corresponding to a cooperative strategy.

In the following, we fixed a and b so that the focal robot will have to invest a value of $\hat{x} = 6.5$ in order to hope to obtain the best possible collective gain, but this gain will only be obtained if her partner consents to the same effort (cf. green curve). However, in the presence of a cooperative partner, it is more interesting for the focal robot to cheat by investing less. In this case, the optimal investment value of our cheating robot is $x_d = 2.5$ (cf. orange curve). Nevertheless, in this case, the partner has no interest in cooperating, and both of our cheaters will eventually get a sub-optimal gain (cf. blue curve). Therefore, the latter situation arises that the only possible Nash Equilibrium is sub-optimal: robots could both get more *and* a robot cannot deviate from this strategy without a loss.

For clarity, we use the following terms in the rest of the Chapter: a robot investing $x_d = 2.5$ will be said to play a *defective strategy*. A robot investing $\hat{x} = 6.5$ will be said to play the *"optimal cooperative strategy"*. Any robot playing $x > 2.5$ will be said to play a *cooperative strategy*, even if it is not the optimal one.

3.2.3 Partner Choice

We give our robots the ability to perform partner choice as a cooperative mechanism to solve the social dilemma we just described. Partner choice makes it possible to escape the sub-optimal selfish behavior of partners by enforcing individuals to act as "good" partners, rather than just optimizing their own self interest. This is made possible by setting up a game during which potential partners announce their respective investment in advance, allowing everyone to decide whether or not to continue the cooperation. As a result, it can supposedly lead to shifting the Nash Equilibrium to the

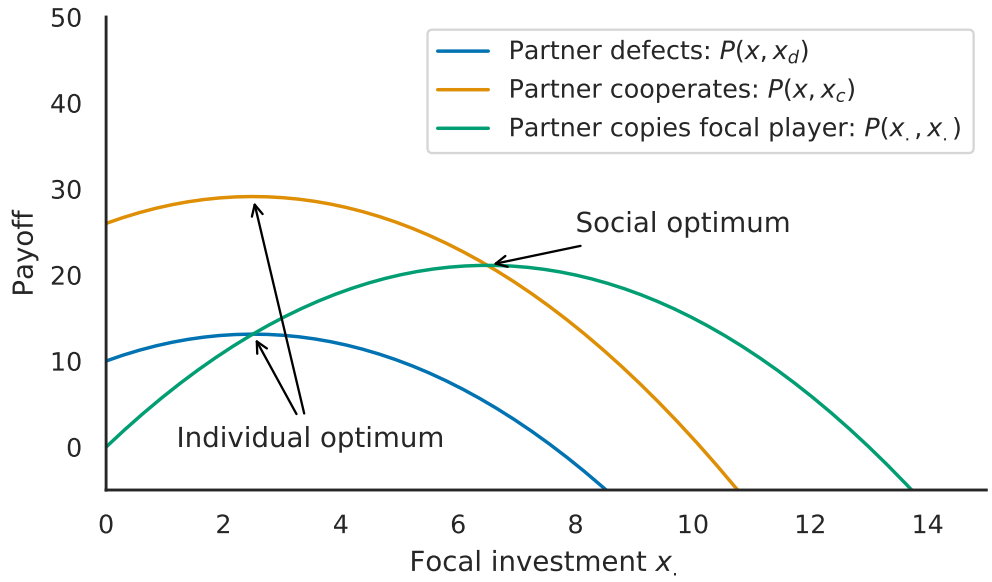

Figure 3.2: Payoff function with different partner's investment value. The individually optimal investment is $x_d = \frac{a}{2}$ whatever the constant value the partner invests, which corresponds to a defective strategy. If both robots invest the same value, then the socially optimal investment is $\hat{x} = a + \frac{b}{2}$, which corresponds to a cooperative strategy behavior.

socially optimal strategy (ie. both partners must cooperate).

Partner choice has been studied in theoretical biology. (Debove, André, et al., 2015) explored learning partner choice in repeated ultimatum games with both field studies with humans and numerical simulations. They showed that the efficiency of partner choice depends on the **meeting probability** of an agent (β) and the **split probability** of an interaction (τ). Both β and τ are expressed from the viewpoint of a focal agent. The meeting probability β determines how fast an agent will find a potential partner (whether it then chooses to interact or not). If the pair interacts, the split probability τ is used to determine when the interaction ends (smaller τ means longer interactions).

If the meeting probability is large compared to the split probability, i.e. β/τ is large, then partner choice is a viable strategy and can emerge. Indeed, for partner choice to be effective, when an agent refuses to interact with a partner, it must do so because its expectation of gain in finding a better partner outweighs the gain missed by rejecting the interaction with the wrong partner and the implied cost paid by looking for a new partner. Thus, if search time is short (ie. larger β) compared to interaction time (ie. smaller τ), it is profitable to spend more time searching for a good partner than interacting with more uncooperative partners.

The β parameter is determined by the ability of the robots to meet on a resource and varies as the robots evolve, but also depending on the density of robots in the arena, and especially the robots that are also seeking for a partner. Both β and τ are fixed in theoretical models. This is not the case in our robotic model, as the meeting probability β is indirectly controlled by how experimental conditions are set (in particular the number of robots and the size of the arena) *and* by how the robots move, which is learned.

3.2.4 Robotic Behaviors

We consider a swarm of heterogeneous robots, meaning that robots may act differently when facing a similar situation depending on their personal learning experience.

Each robot alternates between two behaviors. Firstly, a **foraging** behavior, which is learned. It is in charge of both exploration and partner choice. It will be described in Sections Controller and Representation and Learning; Secondly, a **wandering** behavior, which is hard-coded, that simply moves the robot forward while avoiding obstacles of any kind (walls, resources and other robots alike). This behavior is used *for some time* after a resource has been successfully harvested (see details below).

Partner choice is implemented such that when two robots meet on a resource, each robot executes the following algorithm:

1. The robot announces to its potential partner the effort it is willing to make to capture the resource. This is when the robot can choose to cooperate (to maximize the overall gain) or cheat (to maximize its own gain at the expense of its partner);

2. Based on the effort announced by its potential partner, the robot decides whether to pursue the interaction further;

3. If (and only if) both robots agree to continue the interaction, the resource is harvested and each robot's payoff is computed accordingly. The robot's payoff then depends on both its own investment and the overall investment. As such, two robots may have different payoffs depending on each individual effort.

Whether the interaction is successful or not, the resource disappears and is relocated elsewhere. In case of a successful interaction, both robots switch to the *ad hoc* wandering behavior for a certain period of time which depends on the split probability τ (cf. Section Partner Choice), before returning to its nominal (learned) behavior. The larger the τ value, the faster the robot should start to search for a new resource.

While wandering, the robot no longer participates in the cooperative game as it only avoids obstacles (for example, this corresponds to time taken to process the resource (e.g.: digesting or retrieving the resource). At each time step, each wandering robot has a probability of τ to switch back to the foraging behavior. The wandering behavior is used to simulate the expected duration of an interaction, which value is thus $1/\tau$. τ is fixed for a given experiment, and the influence of several particular values will be explored in Section Results. In the particular case where $\tau = 0$ (ie. interaction time is infinite), then robots that both accepted to interact will be wandering around until the end of the current generation.

3.2.5 Controller and Representation

The robots' control architecture is decomposed, for each robot, into three parts: (1) The **investment value** represents what the robot is willing to pay to cooperate. It is defined in $x \in [0, 10]$; (2) Control parameters for the **partner choice module**. It is used when a robot pairs with another to harvest a resource, to decide whether to accept interaction or not depending on each partner's investment values; (3) Control parameters for the **movement**

Input	Value
Movement module	
Per sensor (×8)	
Distance to Robot	$[0, 1)$ if in range else 1
Distance to Wall	$[0, 1)$ if in range else 1
Distance to Resource	$[0, 1)$ if in range else 1
Partner on the Resource	0 or 1
Partner choice module	
Partner's investment	$[0, 10)$
Robot's own investment	$[0, 10)$

Table 3.1: Neural Networks inputs

module. It is used to move the robot around (e.g. avoid obstacles, finding a resource, finding a partner).

Both modules are artificial discrete neural networks, using a tanh activation function. The details of the inputs of each network are given in Table 3.1. Both networks takes an additional bias value of 1.0 as input. The bias neuron projects on the hidden and output neurons.

The partner choice module is active only when a robot pairs with another on a resource. It takes three input values: the robot's investment value, that of its partner, and a bias neuron. The hidden layer is composed of 3 neurons, and the network produces a single output value ($a \in]-1, 1[$), which determines if cooperation is accepted ($a > 0$) or not ($a <= 0$). The hidden layer is composed of three neurons.

The movement module takes $8*4$ sensory input neurons and a bias neuron as input. These are projected on one hidden layer with 10 neurons. Then, two output neurons are defined in $(-1, 1)$, and scaled to determine the proper translation and rotation speeds. The $8 * 4$ sensory inputs are provided by 8 sensors, placed uniformly around the robots. Each sensor gives four elements of information: (a) distance to nearby robot (if any), (b) distance to wall (if any), (c) distance to resource (if any) and (d) if a resource is detected, presence of another robot on the resource ($= 1$) or not ($= 0$).

3.2.6 Learning

Learning is performed for all individuals, using an evolutionary algorithms as a direct policy search method (Doncieux et al., 2015). Neural weights and investment value are optimized according to an objective function that rewards the ability to forage resources. Each robot is described by its own

unique genome containing 369 values, decomposed as follow: (1) the neural weights of for the movement module, i.e. 352 real values, with each value initialized in the range $[-1, 1)$ and bounded in $[-10, 10]$ throughout learning; (2) the neural weights for the partner choice module, i.e. 16 real values, initialized and bounded similarly; (3) the investment value g_x, a real value defined in $[0, 1)$. At run time, the investment level x of the robot is set to $x = 10 \times g_x$.

The fitness function for each robot used is formalized in Eq. 3.5.

$$F_i = \sum_{j=0}^{n} P(x_i, x_j) \tag{3.5}$$

The fitness value F_i for a given robot i is computed as the sum of its payoffs $P(x_i, x_j)$ obtained during evaluation, with x_i the robot's investment value (which remains constant through evaluation), and x_j the investment of its partner at interaction j^{th}.

Fitness proportionate selection, which can maintain diversity in collective evolutionary robotics setups, is used to build a new population. Mutation is applied to the genome of the selected individuals. Each gene g_k of a robot has a probability $\mu = 0.01$ to mutate. If the gene is selected for mutation, then it has a probability of 0.1 to mutate according to a uniform distribution $\mathcal{U}([-10, 10))$ and a probability of 0.9 to mutate according to a normal distribution $\mathcal{N}(g_i, \sigma)$ with $\sigma = \sigma_w = 0.1$ for the weight genes and $\sigma = \sigma_x = 0.1$ for the investment gene. The new generation then performs the task and the process is repeated for $G = 200$ generations (see Table 3.2 for a list of all the parameters).

3.3 Results

3.3.1 Experimental setup

The environment is a circular arena with a diameter of 400px. The robots are 4px[2] diameter disks. The robots have 8 equally distributed sensors with a range of 96px giving them information about their surroundings, such as the presence of other robots, of a resource or of a wall. The robots move through the environment at a maximum translation speed of 2px/iteration and a rotational speed of 30°/iteration. N robots are spread randomly in the environment and 30 resources are randomly scattered throughout the arena. Each generation lasts $T = 100\,000$ iterations. The environment is represented in Figure 3.1.

[2]px is short for pixels, the basic unit length used in the simulator

Param	Description	Value
Payoff		
a	Public good weight	5
b	Prisoner's dilemma weight	3
Environment		
T	Number of iterations per generation	100 000
G	Number of generations per run	200
	Arena diameter	400px
	Robot size	4px
	Robot sensor range	96px
	Robot max speed	2px/iteration
ω	Number of ressources	30
	Ressource radius	3px
	Ressource footprint radius	10 px
τ	End of interaction probability	
Evolution hyper-parameters		
μ	mutation probability	0.01
σ_w	mutation strength of weight genes	0.1
σ_x	mutation strength of investment gene	0.1

Table 3.2: Experimental parameters

The results presented below are obtained by observing of the final generation (i.e.: 200^{th} generation). We ran 24 simulations per condition in all experiments. The code used to generate all results is freely available[3].

In this Section, we explore whether cooperation can easily be learned, or not. Our hypothesis is that while partner choice should enable cooperative behaviors that is socially optimal, such cooperative behavior may be hindered by other factors such as the number of opportunities to meet other robots.

In the following, the influence of several factors are explored that may facilitate (or not) the emergence of partner choice and cooperation behaviors:

- the effect of population size (Section Learning Cooperation and Population Size). Hypothesis: a robot does not have time to search for a "good" partner if the population is small, as all pairing will be quickly made;
- the effect of the duration of interactions by changing the split probability τ (Section Learning Cooperation and Interaction Length). Hypothesis: exploration will be favored when interactions are long as there is a strong cost to cooperate with a "bad" partner.

Section Learning Cooperation and Population Size also presents the main control experiment, i.e. removing entirely partner choice, to demonstrate that partner choice is indeed mandatory to attain efficient cooperative foraging under the right experimental conditions.

In addition, three control experiments are described that explore the sensibility of results with respect to our particular experimental settings:

Section Effect of Mutation Strength (Control) explores the impact of both weaker and stronger mutation strengths applied on the investment gene σ_x. In particular, a weaker mutation may hinder the possibility to innovate towards better cooperators. We show that this is not the case: results are robust w.r.t. mutation.

Section Population Size vs Generations (Control) explores the influence of the parameters chosen for the evolutionary algorithm. Given a constant evaluation budget, a different balance between the number of generations (which is fixed to 200) and the population size may have an impact. For setups that use a small population, it is possible that better results may be obtained by using more generations. To evaluate this, we adjust the setup with the smallest population to an evaluation budget that matches that of the setup with the larger population, by augmenting the number of generations. We show that the evolutionary algorithm is robust w.r.t. the parameters used.

[3]http://pages.isir.upmc.fr/~bredeche/Experiments/ALIFE2020_coopPC_code.zip

Section Wandering and Relocation (Control) focuses on the possible bias due to the particular implementation of the *ad hoc* wandering behavior. The wandering behavior acts as a diffusion process for the robots, but it is clear that diffusion is neither anisotropic nor provides uniform relocation due to the multiple collisions that can occur with obstacles in the arena. We show that using an unrealistic "teleportation" behavior instead, which ensures pure uniform relocation, actually does *not* change the results obtained before, thus confirming that our particular implementation of the wandering behavior does *not* bias the outcome.

3.3.2 Learning Cooperation and Population Size

To test how partner choice enables the emergence of cooperative behavior, we set $\tau = 0$ and the evaluation duration to $T = 100\,000$. This supposedly corresponds to a favorable setup as robots will benefit from a long search time and a very engaging commitment (i.e. only one pairing is possible) if they accept to interact. Figure 3.3 provides the results for different population sizes, from 50 to 1000 robots.

At $N = 50$, robots plays the defective strategy. With 50 robots in the arena, the robots are unable to meet and sample enough partners to be selective before the end of the generation. Moreover, the robots are racing to find a partner quickly. Indeed, with $\tau = 0$, the more the task advances in time, the fewer robots are available in the arena and thus the more β decreases throughout the evaluation.

On the other hand, robots evolve a cooperative behavior for N sufficiently large as larger population also implies a higher probability of encounters β. With a population of $N = 1\,000$, the average investment level is close to the social optimum.

To validate the importance of partner choice in the evolution of a cooperative behavior, we build a control experiments where we deactivate the robots' ability to know their partner's investment in order to accept or not accept an interaction. In this condition, whatever the number of robots in the environment, the average investment level always converge to x_d, that is a defective behavior (see Fig. 3.4). In this situation, robots have no way to be selective and cannot choose a cooperative robot over a non-cooperative one. Thus, cooperative robots are not preferentially selected as partners and there is no incentive to invest more than the individual optimum.

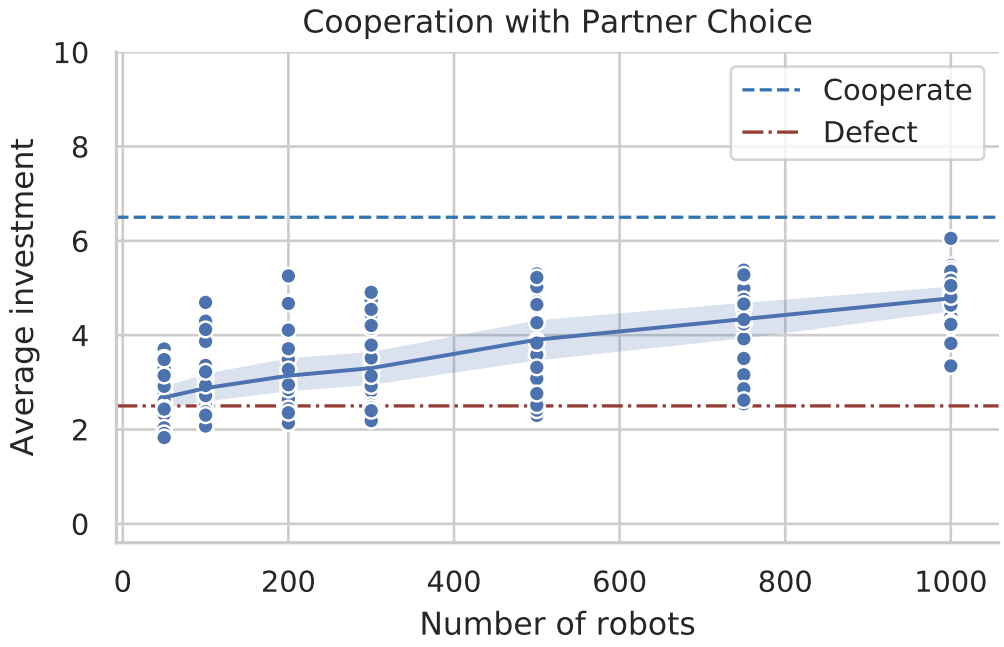

Figure 3.3: Evolution of cooperation with partner choice for a split probability $\tau = 0$ and mutation strength for the investment gene $\sigma_x = 0.1$. For each setup, 24 independent runs are performed (less is shown due to overlaps). Results are compiled from 168 runs obtained from 7 different experimental setups. For each setup, learning is performed for 200 generations with a given population size (x-axis). The values for population size are: 50, 100, 200, 300, 500, 750, 1000. In addition, the blue line shows the average values for each setup with a confidence interval $CI_{0.95}$.

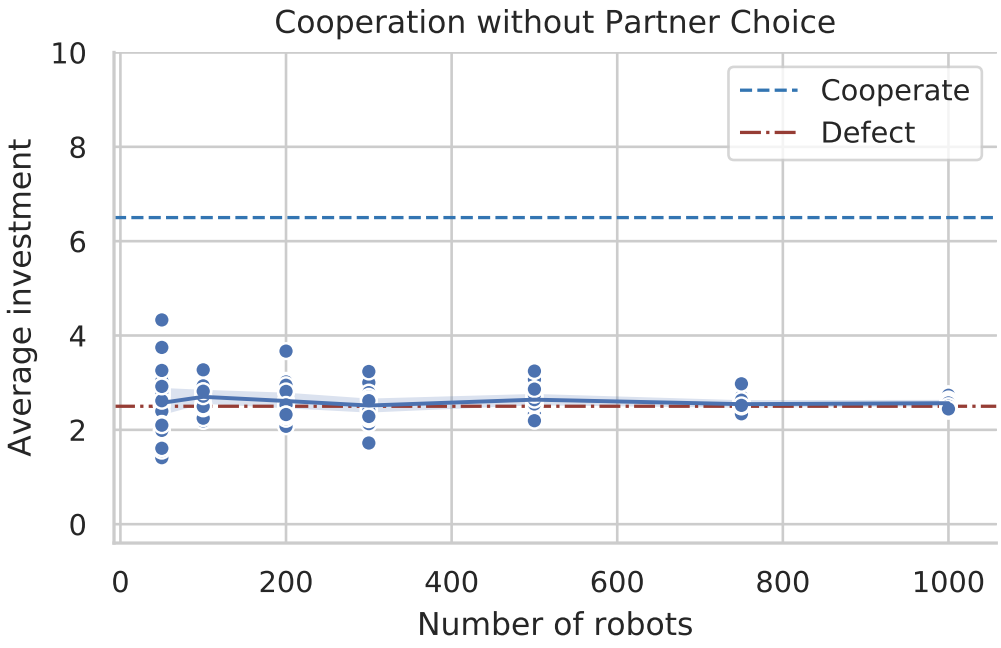

Figure 3.4: Evolution of cooperation *without partner choice* for a split probability $\tau = 0$ and mutation strength for the investment gene $\sigma_x = 0.1$. Technical details are identical to those of Fig. 3.3 (see caption).

3.3.3 Learning Cooperation and Interaction Length

Figure 3.5 shows how the value of the split probability τ affects learning cooperation. When the split probability τ is null or low ($\tau < 10^{-3}$), the robots invest in a collectively optimal way and adopt a cooperative strategy. The robots plays systematically a defective strategy when $\tau \geq 10^{-3}$. Thus, decreasing the split probability (ie. increasing the interaction time) has a positive effect on the acquisition of a cooperative strategy by partner choice, in accordance with previous theoretical results.

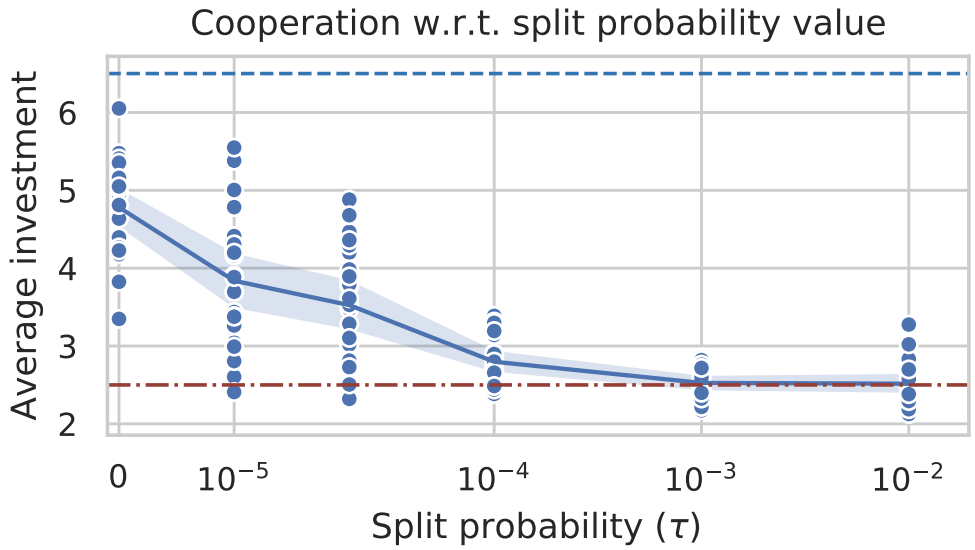

Figure 3.5: Evolution of cooperation with partner choice for a population size of 1000 robots and a mutation strength for the investment gene $\sigma_x = 0.1$. For each setup, 24 independent runs are performed. Results are compiled from 144 runs obtained from 6 different experimental setups. For each setup, learning is performed for 200 generations with a given split probability τ (x-axis). The values for τ are: 0, 10^{-5}, 5×10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}. In addition, the blue line shows the average values for each setup with its 95% confidence interval.

3.3.4 Effect of Mutation Strength (Control)

Previous results in theoretical biology have shown the importance of variability in the level of investment in the population (McNamara et al., 2008). In a population with increased phenotypic diversity, each individual can select partners from a pool of individuals with different strategies. Cooperators may then be chosen over non-cooperators, just because they are available. As a consequence, this can bootstrap the emergence of cooperative strategies.

We test the influence of phenotypic variability (which, in our case, is directly linked to the investment value g_x), induced by smaller or larger mutation rates. To do so, we modify the strength σ_x of the Gaussian mutation on the gene encoding the robot investment level.

Figure 3.6 shows that differences are minor in the average investment level between the different simulations when using larger and smaller mutation strengths (to be compared with the original results in Fig.3.3). However, there is less variability between simulations when the mutation level is high ($\sigma_x \geq 0.1$), which can be explained by a more rapid convergence towards the optimal investment level.

The fact that the variability of investment in the environment plays very little role in our task may be due to the presence of individuals with various levels of investment in the initial population. The ability to be selective in the choice of partner may therefore emerge before the population is completely homogeneous and thus generating phenotypic variability becomes an unnecessary feature.

3.3.5 Population Size vs Generations (Control)

The difference in population sizes between low (50 robots) and large (1000 robots) populations could be explained by the smaller evaluation budget used with small populations. Indeed, given the number of generations is constant ($G = 200$), the number of evaluations when considering a population of 50 robots is $50 \times 200 = 10000$, while a population with 1000 robots gets $1000 \times 200 = 200000$ evaluations. This difference in the number of evaluations could explain why cooperative behavior has evolved in the conditions where N is large and not in those where N is small.

We test the impact of the number of evaluations by running a new control condition of 24 simulations with $G = 4000$ for a population of $N = 50$ robots, offering 200000 evaluations. Fig. 3.7 shows the difference between this new setup ($N = 50, G = 4000$) and both the previous setup with a similar population size but fewer generations ($N = 50, G = 200$) and the previous setups with similar number of evaluations but a larger population

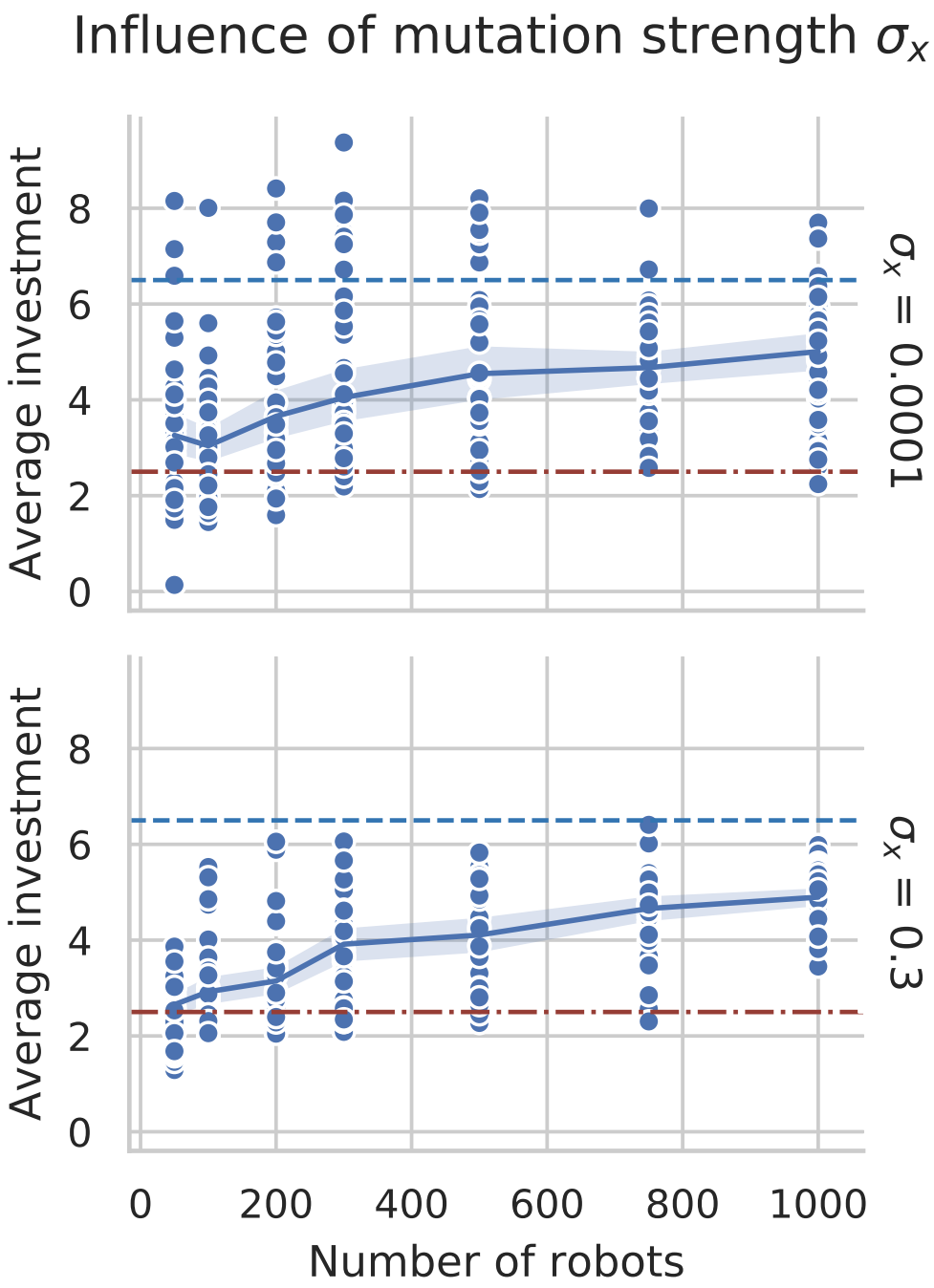

Figure 3.6: Evolution of cooperation with partner choice for a split probability $\tau = 0$ and mutation strength for the investment gene of $\sigma_x = 0.0001$ (top) and $\sigma_x = 0.3$ (bottom). Technical details are identical to those of Fig. 3.3 (see caption), which showed results for $\sigma_x = 0.1$.

$(N = 1000, G = 200)$.

The difference between the $(N = 50, G = 200)$ setup and the $(N = 50, G = 4000)$ setup turns out to be marginal, while the difference with the $(N = 1\,000, G = 200)$ setup using larger population remains largely significant. We conclude that adding more generations does not improve the level of cooperation achieved for conditions with a small population. These results confirm that smaller meeting probability β is responsible for blocking the emergence of cooperative behavior under these conditions.

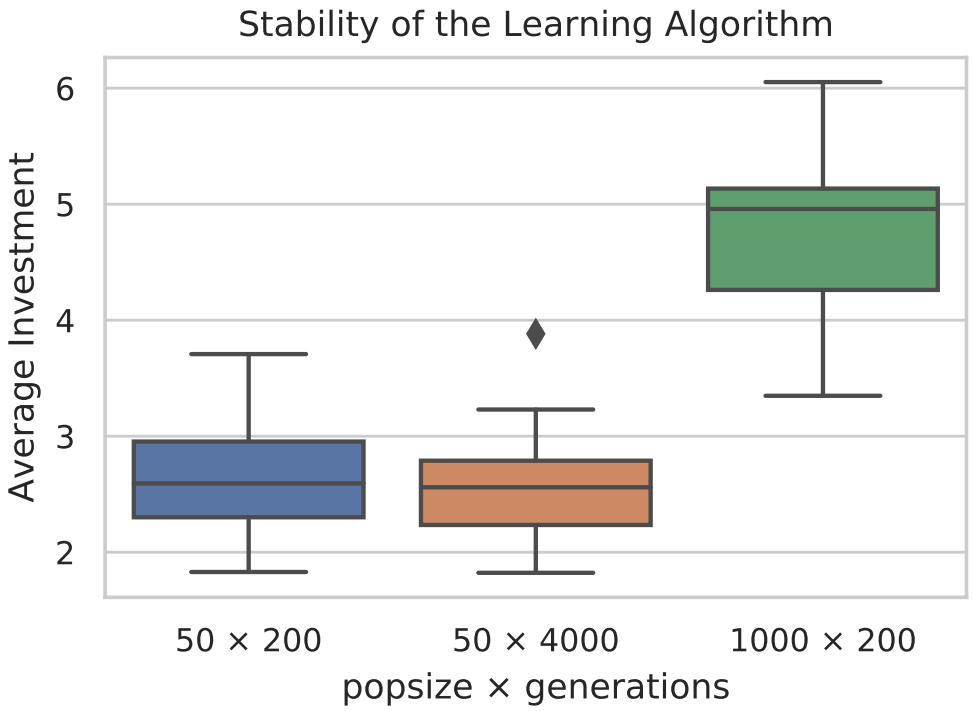

Figure 3.7: Performance is compared using different budget balance between population size and number of generations. From left to right: (1) population=50, generations=200; (2) population=50, generations=4000; (3) population=200, generations=1000. Results from (1) and (3) are taken from Fig. 3.3 and uses different number of evaluations, but the same number of generations (200). Results from (2) are obtained with an evaluation budget similar to (3), i.e. $50 \times 4000 = 1000 \times 200 = 200\,000$, but with a population similar to (1), i.e. 50 robots.

3.3.6 Wandering and Relocation (Control)

In order to account for a possible bias due to the *ad hoc* wandering behavior used, an unrealistic "off-grid" behavior is introduced in place of the wandering behavior. The off-grid behavior is a mechanism that simply removes a robot from the environment after a successful interaction, and relocates it at a random position after some time depending on the split probability τ. It is actually closer to the abstract process used in numerical simulation in evolutionary biology models on partner choice, where space is ignored (Debove, André, et al., 2015).

Figure 3.8 shows results obtained with the off-grid behavior for various values of split probability τ (instead of the wandering behavior, as used for results previously shown in Figure 3.5). Using either the off-grid behavior or the wandering behavior produces similar results. When the split probability is low ($\tau < 1 \times 10^{-5}$), robots tend to be more cooperative (investment value > 2.5). When it is large ($\tau > 10^{-3}$), robots' investment values converge to the 2.5, which means that defection is the rule with either behavior.

Using the off-grid behavior instead of the wandering behavior does actually provide an advantage for intermediate investment values, as robots remain cooperative for larger τ values. This can be explained by the fact that the arena is less crowded than in the wander condition due to the removal of robots from the arena. Indeed, a robot necessarily crosses *only* potential partners, and is not blocked by robots wandering around that cannot be available for interaction. In other words, the β encounter probability is greater with the off-grid behavior than with the wander behavior (see Section Partner Choice).

3.4 Conclusion

We have shown that partner choice in evolutionary swarm robotics with heterogeneous population is a key mechanism to overcome deceptive social dilemma. We have also shown that efficient partner choice can be learned, but that its success strongly depends on environmental conditions. In particular, the number of encounters should be high, and the impact of interaction during or after cooperation should be long in duration.

The scope of the results presented here goes beyond evolutionary swarm robotics, as learning to choose a partner is relevant in other setups, whether this concerns artificial agents (as with on-line distributed learning for solving games) or living individuals (as our model extends models from evolutionary biology and social learning with both learning and embodiement).

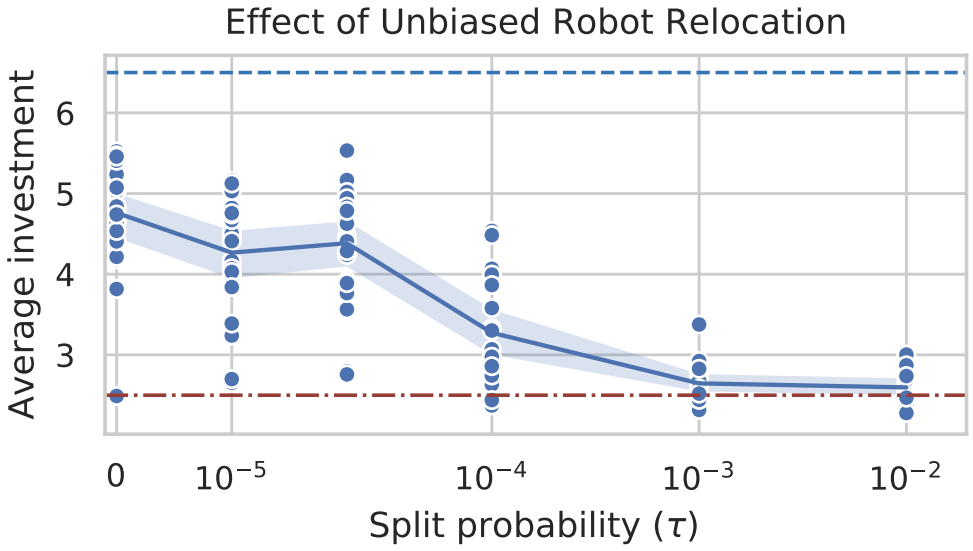

Figure 3.8: Evolution of cooperation *with off-grid behavior instead of the wandering behavior* with a population size of 1000 and mutation strength for the investment gene $\sigma_x = 0.1$. Technical details are identical to those of Fig. 3.5 (see caption).

3.5 Supplementary Materials

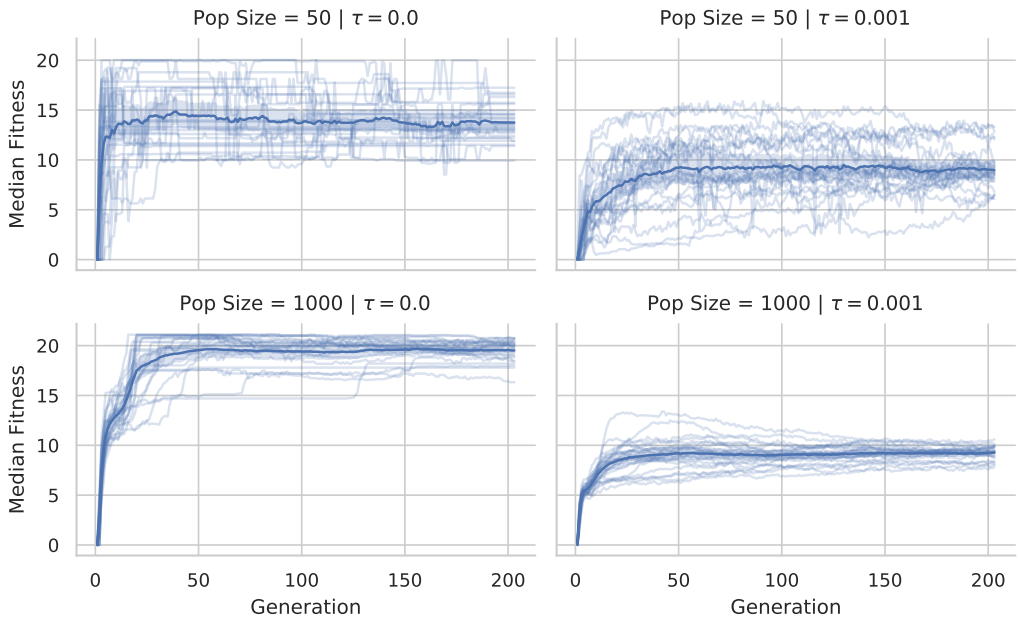

Figure 3.9: Median Fitnesses of all 24 runs for the conditions $N = 50$, $\tau = 0$; $N = 50, \tau = 0.001$, $N = 1000$, $\tau = 0$ and $N = 1000$, $\tau = 0.001$. The Median fitnesses with a small population is much more noiser between the simulations than with a high populatiion. The smaller τ is, the higher is the median fitnesses of the agents.

Chapter 4

Policy Search when Significant Events are Rare: Choosing the Right Partner to Cooperate with

In this chapter we question the very nature of the learning method used to achieve cooperation with partner choice. Previously, we used the same evolutionary algorithm to act both as a process to simulate evolutionary adaptation (Chapter 2) and as an optimisation algorithm in a evolutionary robotics setup (Chapter 3). However, this choice may be questionned when it comes to robotics, especially with the rise of efficient reinforcement learning techniques that can deal with continuous state and action spaces.

We focus solely on the learning algorithm as a blackbox, by comparing two well-established algorithms, the state-of-the-art PPO algorithm for gradient policy search, and the state-of-the-art CMA-ES algorithm for direct policy search. Our motivation is to identify whether these two algorithms yield similar or different results when used in a setting where agents can choose their partner with whom to cooperate. In particular we are interested in identifying fast and efficient algorithms.

To do so, we build from the work presented in the previous chapters, by conserving similar experimental conditions, where independent agents compete with one another to maximise their own gain. However, we simplify the problem by focusing on the learning of only one agent facing partners that follow (different) deterministic strategies. We also set environmental parameters so that the condition for cooperation is optimal (i.e. the agent can play an optimal strategy by learning to cooperate with the "good" partners).

Learning to cooperate with partner choice turns out to be an interesting

reinforcement learning problem: because an agent spends most of its time looking for partners, opportunities for cooperating are actually rare. The main contribution we present in this Chapter is that the rarity of significant events for cooperation may have dire consequences with gradient policy search methods, and not for direct policy search method (which, fortunately, we used in the previous Chapters). The more general message from this work is that in the setup we are interested in, replacing one learning algorithm by another should be done with extreme caution.

4.1 Introduction

This chapter focuses on the evolution of cooperation between unrelated individuals, when each individual can choose whether or not to cooperate with a potential partner. Partner choice implies that each individual chooses whether or not it wishes to cooperate effectively. Thus, individuals will cooperate only if each individual feels that the gain from cooperating exceeds the cost of cooperating. The notion of cost can cover both the effort required to cooperate (e.g., catching prey) and the expectation of a better gain (e.g., finding another, more cooperative partner).

The evolution of cooperation is widely studied in biology, whether in nature (Davies, 2012), in vitro (Kawecki et al., 2012; Lenski et al., 1998), or by means of mathematical models (Murray, 2002) or individual-based models (Railsback & Grimm, 2019; Schulze et al., 2017). When individuals are not genetically related, cooperation is an independent learning problem that is well modeled in competitive evolutionary games: each individual tries to maximize its own gain in relation to others and the environment. When partner choice is possible, previous studies have shown that cooperation is optimal only under certain conditions (Campennì and Schino, 2014; Debove, André, et al., 2015; McNamara et al., 2008 and the preceding Chapters of this thesis).

In particular, the mechanism for partner choice is effective when two conditions are met. First, the number of cooperation opportunities must be high enough that an individual can refuse to cooperate with a potential partner if it hopes to meet a more interesting partner. Second, if an individual and its partner decide to cooperate, the actual duration of this cooperation must be long enough to make cooperation with an uninteresting partner significantly costly. These conditions have an important impact on an individual's experience as the optimal strategy involves rare but guaranteed rewards. Indeed, an individual following the optimal strategy will have to be very demanding in choosing a partner, which can only be possible if the probability of meeting

this "ideal" partner is non-zero.

Beyond the understanding of an important biological mechanism in the evolution of social behaviors, the problem of learning cooperation (with or without partner choice) also arises for artificial agents, whether simulated agents or robots (e.g. in swarm robotics) when learning occurs at the agent level and during the robot's life according to its actions. In this chapter, we are interested in the **independent, on-line and on-policy learning** of cooperation with partner choice.

The tools used in individual-based modeling to model partner choice learning are based on biologically plausible methods, at various degrees of abstraction, simulating the evolution of behavioral strategies. As we consider artificial (robotic or simulated) agents, it is obviously not relevant to be limited to biologically realistic methods. For an artificial agent it is a question of optimizing its own gain according to a succession of interactions with potential partners.

First, the objective of this chapter is to formulate the problem of cooperative learning with partner choice in the formal framework of individual reinforcement learning. This learning problem is characterized by the presence of rare opportunities for non-zero rewards, which is similar to the problem of significant rare events (SRE) studied by Bhatnagar et al. (2006), Ciosek and Whiteson (2017), Frank et al. (2008) in the framework of reinforcement learning.

Second, we evaluate and compare two state of the art methods for on-policy reinforcement learning working in continuous state and action spaces, namely (1) a deep learning method (PPO, Schulman et al., 2017) for gradient policy search and (2) an evolutionary method (CMA-ES, Hansen and Ostermeier, 2001) for direct policy search. We are particularly interested in the influence of the rarity of significant events on the learning dynamics in terms of speed of convergence as well as (and most importantly) in terms of the quality of the policies obtained.

Finally, we will conclude this chapter with a discussion of the algorithmic aspects and the nature of the problem of significant rare events in learning to cooperate.

4.2 Methods

4.2.1 Learning with Rare Significant Events

We consider an independent learner $x_\bullet$, called the *focal agent*, which is placed in an aspatial environment. At each time step, $x_\bullet$ is presented with either

a *cooperative partner* $x_i^+ \in X^+$ or a *non-cooperative partner* $x_j^- \in X^-$. X^+ (resp. X^-) is the finite set of all cooperative (resp. non-cooperative) agents, with both i and $j \in \mathbb{N}$ and > 0. The outcome of the interaction of the focal agent with its partner will depend on its own action and that of its partner. When presented with a cooperative partner x_i^+, the focal agent's reward can be non zero (see Section 4.2.2 for details). When presented with a non-cooperative partner, the reward will always be zero.

Our objective is to endow the focal agent $x_\bullet$ with the ability to learn how to best cooperate, which implies to negotiate with its potential partners and decide whether cooperation is worth investing energy in, or not (see Section 4.2.3 for details). The focal agent faces an individual learning problem as it must optimize its own gain over time in a competitive setup: for cooperation to occur between the focal agent and a partner, the partner must willing to cooperate (ie. be one of x_i^+) and both the focal agent *and* the cooperative partner must estimate that one own energy invested in cooperation is worth the benefits.

We use the standard reinforcement learning framework proposed by Sutton and Barto (2018) to formalize the learning task from the focal agent's viewpoint, which is essentially a single agent reinforcement learning problem.

The focal agent $x_\bullet$ interacts with the environment in a discrete time manner. At each time step $t = 0, 1, 2, ...,$ $x_\bullet$ is in a state $s \in \mathbb{R}$ which describes its current partner's investment value, and plays a continuous value $a \in \mathbb{R}$ which represents its decision to cooperate ($a > 0$) or not ($a <= 0$).

Let π_θ be the parametrised policy of the focal agent, with $\theta \in \mathbb{R}^n$. The learning task is to search for θ^*, such as:

$$\theta^* = \underset{\theta}{argmax} J(\theta) \tag{4.1}$$

With J the global function to be optimized, defined as:

$$J(\theta) = \mathbb{E}\sum_t r_t \tag{4.2}$$

with reward r_t (or return) at time t. Rewards are defined such that $r \in \mathbb{R}$ and depends on the current state s and action a, and are produced according to the probability generator defined as follow:

$$r(s,a) = \begin{cases} payoff(s,a) & \text{with probability } p \\ 0 & \text{otherwise.} \end{cases} \tag{4.3}$$

The probability $p \in [0, 1]$ determines the probability to encounter a cooperative agent (i.e. one of x_i^+). The value of p depends on the setup, and

determines how *rare* non-zero rewards may occur when $p < 1.0$. A probability of $p = 1.0$ means the focal agent $x_\bullet$ encounters a cooperative partner at each time step t, with a possible non-zero reward that depends on the *payoff* function. Non-zero rewards become rarer (but still possible) as $p \to 0$. Note that $\mathit{payoff}(s, a)$ is non-zero *only* if both the focal agent *and* its cooperative partner accept to cooperate. Cf. Section 4.2.3 for details on the negotiation process.

The problem presented here is very similar to that of Rare Significant Events as formulated by Frank et al. (2008). However, our problem differs on two aspects. Firstly, we consider on-line on-policy search of a parametrised policy, where the frequency of rare events cannot be controlled. Secondly, and even more importantly, a learning episode stops right after the focal agent and a cooperative agent have reached a consensus to cooperate, i.e. when $\mathit{payoff}(s, a) \neq 0$. If no cooperation is triggered, an episode stops after $T = \frac{100}{p}$ time steps. Therefore, the expected number of meetings (M) is held constant, $E(M) = 100$ as p varies.

The situation that is modelled here corresponds to many collective tasks observed in nature (Bshary & Noe, 2003; Simms & Lee Taylor, 2002; Wilkinson et al., 2016), where each agent has to balance between looking for partners and cooperating with the current partner, the latter possibly taking significant time. As a matter of fact, it has been shown elsewhere (Campennì & Schino, 2014; Debove, André, et al., 2015; McNamara et al., 2008), as well as in the previous Chapters, that optimal partner choice strategies can be reached only when the cost of cooperation is large (ie. the duration of cooperation is long with regards to looking for cooperative partners).

4.2.2 Partner Choice and Payoff Function

Whenever the focal agent $x_\bullet$ and a cooperative partner x_i^+ interact together, they play a variation of a continuous Prisoner's Dilemma. Cooperation actually takes place if *both* agents deem it worthwhile. The procedure for partner choice is the following:

- Investment module: each agent simultaneously announce the investment they are willing to pay to cooperate;
- Choice module: each agent then decides to continue the cooperation based on the investment announced by its partner and its own.

To simplify notations, we use $x_\bullet$ and x_i^+ to represent both the individuals and the investment values they play, i.e. $x_\bullet$ (resp. x_i^+) plays $x_\bullet$ (resp. x_i^+). The gain received by the focal agent $x_\bullet$ is defined as:

$$P(x_\bullet, x_i^+) = a \times x_\bullet + b \times x_i^+ - \frac{1}{2}x_\bullet^2 \tag{4.4}$$

With $a, b \geq 0$ and $a + b > 0$. This payoff function combines both a prisoner's dilemma and a public good game, and was first introduced in Chapter 2. Two different Nash equilibria can be reached for $x_\bullet$:

- $x_d = a$. This is a sub-optimal equilibrium, which corresponds to an agent cheating, a typical outcome in the prisoner's dilemma where an agent maximizes its own gain, but also minimizes its exposure to defection. This ensure the best payoff for the agent if it is unable to distinguish a cheater from a cooperator.

- $x_c = a + b$. This is the optimal equilibrium, where both agents are play cooperatively to maximize their long-term gain.

The public good game is included in the payoff function to help distinguish between agents that are simply ignoring the cooperation game $(x_\bullet) = 0$), from those who takes part of it, even if they defect $(x_\bullet) \geq x_d$).

The focal agent can get the optimal payoff if it plays $x_\bullet = x_c$ *and* its partner plays $x_i^+ = x_c$, which can occur if particular conditions are met when partner choice is enabled. Partner choice can lead to optimal individual gain whenever a successful cooperation removes the possibility for further gain with other partners. In other words: the focal agent can meet with any number of possible partners but will take the gain of the first and single mutually accepted cooperation offer.

In this chapter, we set $a = 5$ and $b = 5$, therefore $x_d = 5$ and $x_c = 10$. The maximum payoff the agent can obtain is to invest $x_\bullet = x_c$ with its partner investing equally $x_i^+ = x_c$. In this context, $P(x_\bullet, x_i^+) = 50$. The focal agent's investment is bounded as $0.0 \leq x_\bullet \leq 15.0$. This is similar for x_i^+.

$P(x_\bullet, x_i^+)$ and *payoff*(s, a) (introduced in Equation 4.3) differs as the P function relates to the game theoretical setting while the *payoff* function relates to the reinforcement learning problem. On the one hand, the *payoff* function computes the focal individual's reward whether *or not* cooperation was initiated. On the other hand, P computes the focal individual's gain that results from a cooperation game between two individuals that *accepted* to cooperate. However, both functions are linked. From a notational standpoint, s represents the investment values of the focal individual $x_\bullet$ and of its partner x_i^+, and a represents the decision to cooperate and depends on s. The return value of *payoff*(s, a) depends on whether cooperation was initiated or not. If both agents decided to cooperate, then the focal agent's payoff is *payoff*$(s, a) = P(x_\bullet, x_i^+)$, with $P(x_\bullet, x_i^+) \leq 50$ in this case. If cooperation

fails, the focal agent's payoff is $payoff(s, a) = 0$ (which is obtained without having to compute P). The *payoff* function in Equation 4.3 can be written as follow, with updated notations and assuming $a_\bullet > 0$ (resp. $a_i^+ > 0$) means the focal agent (resp. partner) is willing to cooperate:

$$payoff(s_\bullet, a_\bullet) = \begin{cases} P(x_\bullet, x_i^+) & \text{if } a_\bullet > 0 \text{ and } a_i^+ > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{4.5}$$

4.2.3 Behavioural Strategies

For each interaction, the focal agent's investment value $x_\bullet \in [0, 15]$ is computed, and when the investment value of its partner is known, its decision to cooperate $a_\bullet \in \mathbb{R}$ is computed to determine if cooperation should be pursued or not. Each value is provided by a dedicated decision module:

- the **investment module** which provides the cost $x_\bullet$ that the focal agent is willing to invest to cooperate. This module takes no input as it is endogenous to the agent (i.e. the cost $x_\bullet$ is learned, not computed);

- the **choice module** takes both the focal agent's own investment value ($x_\bullet$) and that of its partner (x_i^+ or x_j^-), and computes $a_\bullet$, which is used to determine if cooperation is an interesting choice ($a_\bullet > 0$) or not ($a_\bullet \leq 0$). The choice module is essentially a function $f_{choice}(x_\bullet, x_{partner}) \to a_\bullet$ with $x_{partner} \in X^+ \cup X^-$. The parameters of the function are learned, and the decision to cooperate is computed (as the decision to cooperate is conditioned by the partner's investment).

With respect to the focal individual, Section 4.3 describes how the investment and choice modules are defined and how learning is performed depending on the learning algorithm used.

Cooperative partners x_i^+ and non-cooperative partners x_j^- also use similar decision modules, providing investment and choice values. However, all use deterministic fixed strategies, which may differ from one partner to another. Firstly, non-cooperative partners x_j^- all follow the same strategy. Both the investment value x_j^- and the decision to cooperate a_j^- are always 0, $\forall j$.

Secondly, cooperative partners x_i^+ follow stereotypical but diverse cooperative strategies depending on the value i. Each cooperating partner invests a fixed value $x_i^+ \in [0, 15]$ defined as:

$$x_i^+ = \frac{i-1}{i_{max}} \times 15,\ i \in \{1, \ldots, i_{max}\} \tag{4.6}$$

Each cooperative partner then accepts to cooperate if the focal agent's investment value $x_\bullet$ is greater or equal to their investment, which is written as follow:

$$a_i^+ = \begin{cases} 1 & \text{if } x_\bullet \geq x_i^+ \\ -1 & \text{otherwise.} \end{cases} \tag{4.7}$$

In the following, there are $i_{max} = 31$ cooperating partners ($x_i^+ \in X^+, i \in \{1, \ldots, 31\}$).

4.3 Parameter Settings and Algorithms

We use two reinforcement learning algorithm: a gradient policy search algorithm and a direct policy search algorithm. Both algorithms are used to learn the parameters of the focal agent's decision modules.

4.3.1 Proximal Policy Optimization

The deep reinforcement learning Proximal Policy Optimisation (PPO) (Schulman et al., 2017) is a variation of the Policy Gradient algorithm (Sutton & Barto, 2018). Policy gradient algorithms maximise the function that maps the policy parameters θ of the policy π to the expected reward of this policy on a episode $J(\theta)$, defined in Eq. 4.2.

Though, as the expected value of a certain state-action pair varies according to the policy itself, updating a new policy from samples acquired from an old policy may cause false predictions, as the expected value of an action-state pair may be wrong with respect to the new policy. PPO ensures that the policy generated from the samples of the old policy does not differ too much from the old policy. This ensures the stability of the learning process.

As we are dealing with episodes and do not want to encourage our agent to act in the least amount of time steps as possible, the discount factor is to $\gamma = 1.0$, as recommended by Sutton and Barto (2018, p.68). We used a grid search on the learning rate lr, the minibatch size, the number of SGD iterations and the batch size to find the best parameters. After this grid-search, we picked $lr = 0.005$. The number of SGD iterations, batch size and mini-batch size we tested had no impact on the performance at the end of the learning nor notable speed up in convergence. The hyper-parameters are reported in table 4.1.

The investment and decision modules are both represented as Artificial Neural Networks (ANN). A module is composed of both a controller and a value function, as PPO runs as an actor-critic algorithm. The value function

Parameters	Values
Learning rate	0.005
Optimiser Algorithm	SGD
Number of optimisation epoch	10
Minibatch size	128
Batch size	4000
Kullback-Leibler coefficient	0.2
Kullback-Leibler target	0.01
GAE Parameter λ	1.0
Discount factor γ	1
Value Function loss coefficient	1
Entropy coefficient	0
Number of weights	36

Table 4.1: Parameters used for the PPO algorithm

network has the same layout as the controller, but only output one scalar, the value of the state. The controller from the investment module is a simple neural network with one dummy input (always 0.0, due to implementation constraints of the library used) and a bias (always 1.0), no hidden layer and two outputs, the investment mean m and standard deviation σ. The investment $x_\bullet$ is picked along the distribution $\mathcal{N}(m, \sigma^2)$ and clipped between 0 and 15. This continuous stochastic action selection allows the PPO agent to explore. The controller neural network has 4 weights, and the value function has 2 weights. In total, the investment module is composed of 6 weights.

The partner choice neural network controller has 2 inputs and a bias, 3 neurons and a bias in one hidden layer and has two outputs, accept or refuse. A softmax probabilistic choice is done to choose which action to make. In total, with the value function estimator, 30 weights compose this module. The activation functions for the intermediate layers are tanh and the activation functions for the output layers are linear for both models.

4.3.2 Covariance Matrix Adaptation Evolution Strategy

The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is an optimisation algorithm that does black box optimisation and is derivative-free (Hansen & Ostermeier, 2001). The goal of CMA-ES is to find x^* that maximises (or minimises) a continuous function f. CMA-ES does not require

the function to be convex or differentiable. CMA-ES relies on stochastic sampling around a guess. CMA-ES creates a population of size λ around the guess using a multivariate Gaussian distribution. Each individual of the population is evaluated and CMA-ES then creates a new guess based on the average of the individuals weighted by their evaluation rank. Furthermore, the covariance matrix of the multivariate Gaussian distribution is updated so that the distribution is biased toward the direction where the best solutions were from.

Both decision modules (investment and partner choice) are represented as ANN. A module is composed solely of a controller, CMA-ES does not rely on value function. The controller from the investment module is a simple neural network with one dummy input (always 0.0) and a bias, no hidden layer and 1 output, the investment of the agent, which is clipped between 0 and 15. The controller neural network has 2 weights.

The partner choice neural network controller has 2 inputs and a bias, 3 neurons and a bias in one hidden layer and has two outputs, accept or refuse. A softmax probabilistic choice is done to choose which action to make. In total, 17 weights compose this module. The activation functions for the intermediate layers are tanh and the activation functions for the output layer are linear for both models.

We optimize both modules as a single vector of weights. To have comparable search space dimension with PPO, dummy weights were added to the vector. We choose $\sigma_{init} = 1$ for the standard deviation at the beginning of the simulation and a vector of zeros as guess. The population size λ for the model was chosen by the following rule, which is the default population size in the python CMA-ES implementation (Hansen et al., 2020), be N the number of dimension in the model, λ is computed with the formula given in Eq. 4.8.

$$\lambda = 4 + \lfloor 3 \times \ln(N) \rfloor \tag{4.8}$$

We encode the weights of both modules into one genome composed of 34 genes (a "candidate solution" of the focal agent). The population size of CMA-ES is therefore $\lambda = 14$. Once the λ candidate solutions have been evaluated, a new population is generated according to their performance. Therefore, a new population is generated every 14 episodes, and so forth until the evaluation budget is consumed.

Parameter	Value
population size	14
σ_{init}	1
Number of weights	34
Number of episode per evaluation	1

Table 4.2: CMA-ES parameters table

4.4 Results

The environment, the models and the learning algorithms are implemented with ray[1], rllib[2] and pytorch[3]. We use the cma[4] package in python for the CMA-ES implementation. Source code is available at https://github.com/PaulEcoffet/RLCoopExp/releases/tag/v1.

For a given value of probability of rare events p, we performed 24 independent runs for each algorithm. A run lasts 200 000 episodes. The maximum duration of an episode is fixed as described in Section 4.2.1 so the expected number of significant events remains identical. In practical, an episode lasts *at most* 100 (resp. 200, 500, 1000) iterations for $p = 1.0$ (resp. 0.5, 0.2, 0.1)).

Performance of the current policy is plotted every 4000 iterations, which correspond to the batch size used by *PPO* for learning. As episodes last significantly shorter than 4000 iterations this means the policy's performance is averaged. For CMA-ES, we extract the best individual of the current generation and re-evaluate it 10 times (i.e. for 10 episodes) to get a similarly averaged performance. Results are shown on figures with a data point every 1000 episodes.

4.4.1 Learning with always significant events

Figure 4.1 and Figure 4.2 show the performance throughout learning with the PPO algorithm and the CMA-ES algorithm when $p = 1.0$ (i.e. the focal agent faces only cooperative partners). Each Figure shows 24 curves corresponding the 24 independent runs. Both PPO and CMA-ES are shown to learn near optimal policies (*performance* $\rightarrow$ 50) in almost all runs. CMA-ES is faster to converge than PPO, but offers less robustness as 20 (out of 24) runs with

[1]https://docs.ray.io/en/master/
[2]https://docs.ray.io/en/master/rllib.html
[3]https://pytorch.org/
[4]https://pypi.org/project/cma/

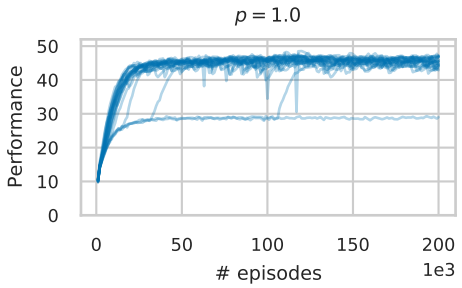

Figure 4.1: Performance of the best policy throughout learning with PPO, for each of the 24 independent runs. There are 23/24 runs that produced a policy where $performance > 40$. The optimal value for a policy can be 50.

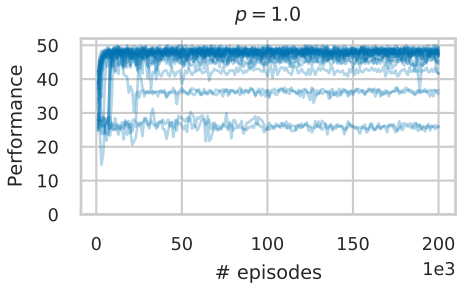

Figure 4.2: Performance of the best policy throughout learning with CMA-ES, for each of the 24 independent runs. There are 20/24 runs that produced a policy where $performance > 40$. The optimal value for a policy can be 50.

CMA-ES reach a performance above 40, to be compared to 23 (out of 24) runs with PPO.

In order to better compare the quality of the policies learned by each algorithm, the best policy from the last population of each run is selected and re-evaluated for 1000 extra episodes without learning. Results are shown in Figure 4.3 with both methods faring similar performance. The mean value for CMA-ES (44.42±6.62) is only slightly less than that of PPO (44.79±3.57), and while the difference between the two distribution may be interpreted as significant ($0.01 < p - value < 0.05$, Mann-Whitney's U-test), the effect size is peculiarly low ($d = 0.07$, Cohen's term). Therefore, it is safe to conclude that both algorithms provide excellent and comparable results when $p = 1.0$.

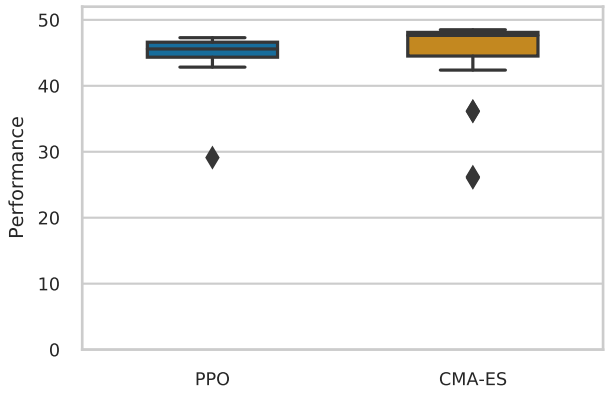

Figure 4.3: Performance of the best policies from PPO and CMA-ES with $p = 1.0$ after re-evaluating policies for 1000 episodes without learning. Two-tailed Mann-Whitney U-test, $n = 24$, $p - value = 0.018$, Cohen's effect size is very small $d = 0.07$.

4.4.2 Learning with rare significant events

Figure 4.4 and Figure 4.5 show the performance (average and 95% confidence interval) of the agent throughout its learning with the PPO algorithm and the CMA-ES algorithm for different conditions of rare significant events ($p \in \{0.1, 0.2, 0.5\}$), as well as with the control condition when all events are significant ($p = 1.0$, copied from the previous Section). Each figure shows the mean performance of 24 independent runs per conditions, compiling each setup by tracing the average performance and 95% confidence interval from the 24 runs.

CMA-ES appears to be only marginally impacted when significant events become rarer (i.e. $p < 1.0$), with all setups showing convergence towards a similar performance value close to the optimal. PPO, on the other hand, is largely affected when significant events are rare (e.g.: average performance of 25 when $p = 0.1$, which is half the theoretical optimal value).

Figure 4.6 shows the results for the additional analysis where the best policy from each run for each condition $p \in \{0.1, 0.2, 0.5, 1.0\}$ is selected and re-evaluated for 1000 extra episodes without learning and with the condition $p = 1.0$ (i.e. only significant events matter here). Results confirm that the difference in the performance of policies obtained with CMA-ES and PPO widens as rewards become rarer ($p < 1.0$) with PPO faring significantly worse than CMA-ES ($p - value < 0.0001$, Mann-Whitney U-test).

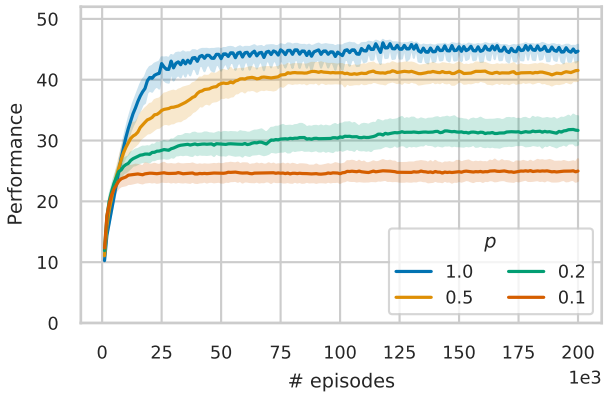

Figure 4.4: Performance of the best policies (average and 95% confidence interval) throughout learning with PPO for the 3 conditions with rare significant events ($p \in 0.1, 0.2, 0.5$) and 1 control condition ($p = 1.0$, cf. also Fig.4.1).

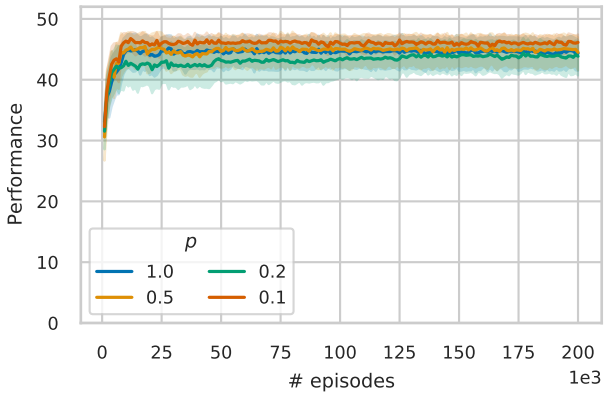

Figure 4.5: Performance of the best policies (average and 95% confidence interval) throughout learning with CMA-ES for the 3 conditions with rare significant events ($p \in \{0.1, 0.2, 0.5\}$) and 1 control condition ($p = 1.0$, cf. also Fig.4.2).

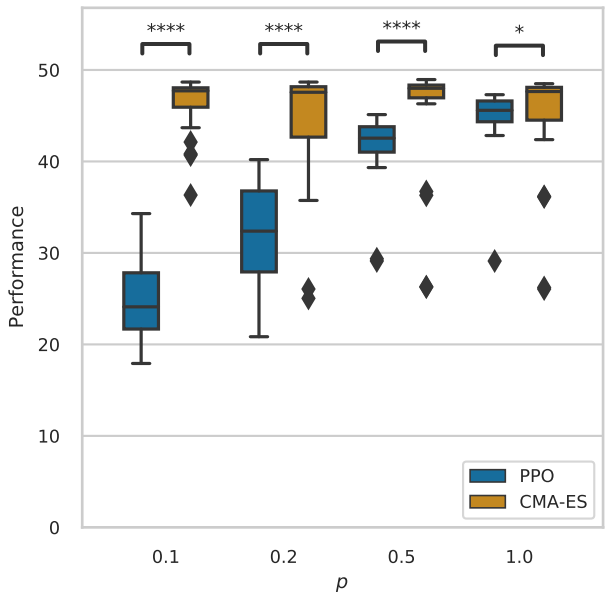

Figure 4.6: Performance of the best policies from PPO and CMA-ES with $p \in \{0.1, 0.2, 0.5, 1.0\}$ after re-evaluating policies for 1000 episodes without learning. Two-tailed Mann-Whitney U-test, $n = 24$ marked as: * for $p - value < 0.05$, ** for $p - value < 0.01$, *** for $p - value < 0.001$ and **** for $p - value < 0.0001$.

4.4.3 Analysing best policies for partner choice

In order to better understand why policies' performance differ among learning algorithms and conditions, the agent's policy obtained at the end of each run is extracted and analysed (i.e. 24 policies per algorithm per condition).

Figure 4.7 illustrates the outcome of the Investment Module ($x_\bullet$), i.e. the investment value offered by the focal agent when faced with a potential partner. It is obtained by measuring the investment value of the focal agent[5] from 1000 episodes with $p = 1.0$ and without learning. Policies learned with CMA-ES play close to $x_c = 10$, which is the optimal play for the payoff function (Section 4.2.2), whatever the frequency of significant events. As expected, this is different for policies learned with PPO, as the outcome values of the Investment Module are significantly lower when the frequency of significant event decreases ($p < 1.0$).

Figure 4.8 illustrates the investment values played by cooperative partners, when the focal agent accepts to cooperate (whether or not cooperation will actually take place, as it also depends on the partner's acceptance). In other words, it represents how demanding is the focal agent with respects to its partners' intention to invest in cooperation. The probability to accept cooperation is computed for the policies of each run. Each policy is presented with all 31 possible cooperative partners, 100 times each, to estimate the focal agent strategy. While CMA-ES produced consistent policies that follow quasi-identical strategies for all conditions (ie. accepting partners that invest close to the optimal $x_c = 10$ or above), this is not the case for PPO policies which are less demanding for lower value of p (with many of the policies learned with condition $p = 0.1$ actually accepting *any* partners).

Figure 4.9 takes a detailed look at the results shown in Figure 4.8. It shows the strategy profile for partner choice by the *best* policy obtained with each algorithm in each condition. Focal agents obtained with CMA-ES follow an efficient and clear-cut strategy: they play the optimal investment value ($x_\bullet = x_c = 10$, green vertical line) and accept partners only when those play a similar or better value ($x_i^+ \geq 10$, blue line). Focal agents obtained with PPO display a comparable strategy for $p = 1.0$ and $p = 0.5$, and less efficient strategies for lower values of p, with both sub-optimal investment values ($x_\bullet < x_c$) and accepting cooperation with partners that invest less than one's own investment ($x_\bullet > x_i^*$).

[5]Note that for CMA-ES the Investment Module follow a deterministic policy (but not the Choice Module). Therefore, it would have been equivalent to take the investment value from the policy parameters in that particular case

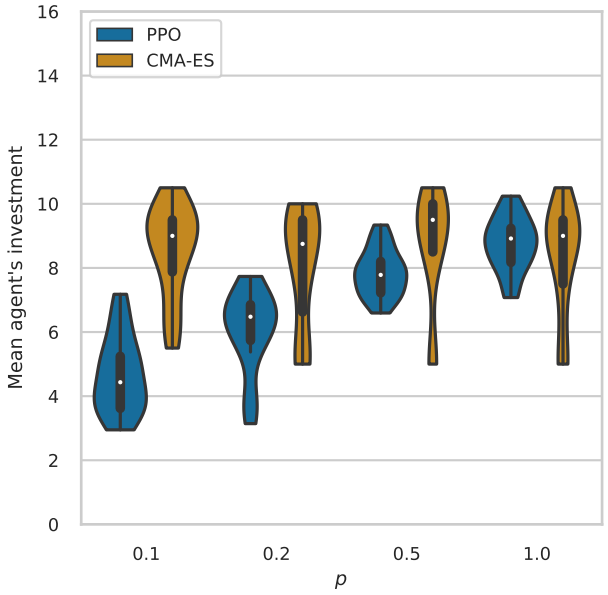

Figure 4.7: Investment value of the focal agent given by the Investment Module for the best learned policies with PPO (blue) and CMA-ES (orange) algorithms, for each condition p. Each violin graph represents the results of the outcome of the 24 best policies for a given algorithm and condition after being re-evaluate for 1000 episodes without learning.

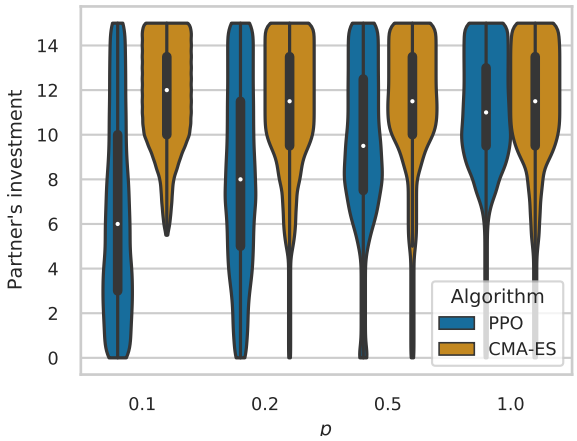

Figure 4.8: Decision to accept to cooperate taken by the focal agent, when facing a cooperative partner with a particular investment value. Results for PPO (blue) and CMA-ES (orange) are shown as violin graph. X-axis: algorithms and conditions, Y-axis: partner's investment value for which the focal agent accept to cooperate.

4.5 Concluding Remarks

In this chapter, we focused on an on-policy reinforcement learning problem of an autonomous agent that needs to maximise its gain when interacting with other agents, with whom our agent may or may not decide to cooperate. The peculiarity of this problem is to present a (very) small number of significant events during which the agent can obtain a non-zero reward. The challenge is therefore to learn how to best choose a partner, by making a compromise between the probability of meeting a potential partner, and the cost of an interaction.

We have studied two reinforcement learning methods: one from the family of gradient policy search and the other from the family of direct policy search. Both allowed learning an optimal use of partner choice when interaction opportunities are frequent. However, the two algorithms differ fundamentally when interaction opportunities are rare. On the one hand, the direct policy search algorithm shows total robustness, while on the other hand, the gradient policy search algorithm collapses, resulting in sub-optimal policies when interaction opportunities are rare.

The robustness of the direct policy search method is expected, as the sequential and temporal aspects of the task is lost within one evaluation: as long as the evaluation time is long enough to sample the whole population

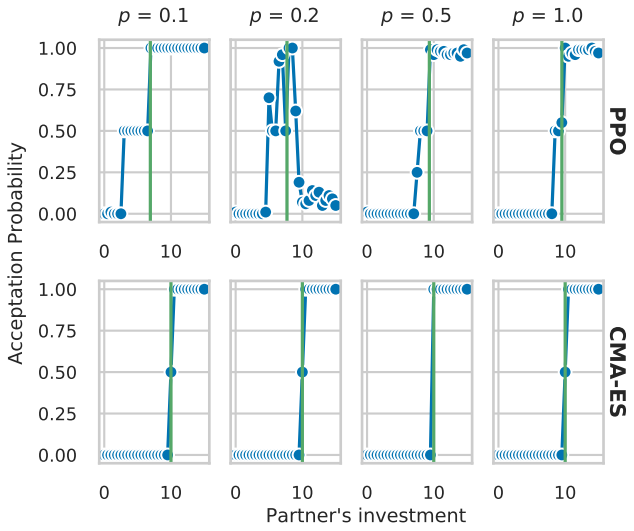

Figure 4.9: Analysis of the Partner Choice module for all conditions (by columns: $p \in \{0.1, 0.2, 0.5, 1.0\}$) and all algorithms (top-line: PPO, bottom-line: CMA-ES). in the condition $p = 1.0$ for PPO. For each setup, only the best policy is shown. Each graph plots the probability to accept cooperation for the focal agent following the best policy (y-axis) depending on its partner's proposed investment (x-axis). Data are computed by presenting each of the 31 possible cooperative partners to the focal agent for 100 iterations as policies are stochastic. The green vertical line represent the mean investment of the focal agent.

of relevant partners, there is no cost nor change in the algorithm dynamics to deal with a situation where significant events are lost within a longer sequence, but still of the same number. This is of course different for the gradient policy search method, where increased rarity means that many learning steps will be performed with zero-reward. Not only this slows down learning, even with a similar number of iterations, but also appears to hinder the ability for the learning to converge towards a truly optimal partner choice strategy. In policy gradient method such as PPO, and contrary to direct gradient-free policy search method such as CMA-ES, the very length of the sequence of events actually matters.

Previous works from Frank et al. (2008) revealed and studied this problem of significant rare events. However, these works were conducted with off-line off-policy reinforcement learning and proposed other methods, such as importance sampling (Ciosek & Whiteson, 2017), to address the problem of rare significant events. In our particular case, such methods cannot be applied because of the on-line on-policy nature of the cooperation problem. It would be interesting though, to endow policy gradient methods with similar mechanisms to stand robust in this context.

In particular, there are possibly some relations to establish between the problem of dealing with rare significant events and that of dealing with sparse rewards, which has gain a lot of attention recently (Jaderberg et al., 2017; Konidaris & Barto, 2006; Riedmiller et al., 2018). In our setup, significant events may be rare, but eventually occur. This may not be the case in a more realistic setup where agents or robots have to move around to meet with another before they can negotiate to cooperate. In that case, rewards would also be conditioned by some distance metric from one agent to another. However, this may not be a problem with cooperation, as we showed in the previous Chapter: encounters may occur occasionally by following a random walk pattern in a bounded 2-dimensional environment (which we actually assumed in the current work). Such a navigation strategy actually results in rare, but not sparse, significant events. This of course may not be the case if the environment is larger and/or more complex, where more complex navigation strategies should be learned.

As a final remark, we highlight that our contribution is two-fold. First, we provided an insight as to how different policy search methods produce different outcomes in an on-line on-policy reinforcement learning problem with significantly rare events. Second, it answers a question related to the biological modelling of the evolution of social behaviours. While algorithms designed for artificial learning and optimisation may seem similar from afar, and could possibly be used to study the dynamics of adaptation, the devil is in the details: the evolutionary (or learning) dynamics resulting from a given

method strongly depends on its implementation details.

4.6 Supplementary Materials

4.6.1 Detail analysis of the agents' reward

We plot the performance per run of each conditions in Fig. 4.10. As p gets lower, the performance reached by a run gets slightly lower. Furthermore, for $p \in \{1, 0.5\}$, agent's reward converges to two different values: one optimal equilibrium above 40 and a second sub-optimal equilibrium at around 30. There are two equilibria that the algorithm reaches. The transition from the sub-optimal equilibrium to the optimal equilibrium is quick, but occurs less and less often as p gets smaller.

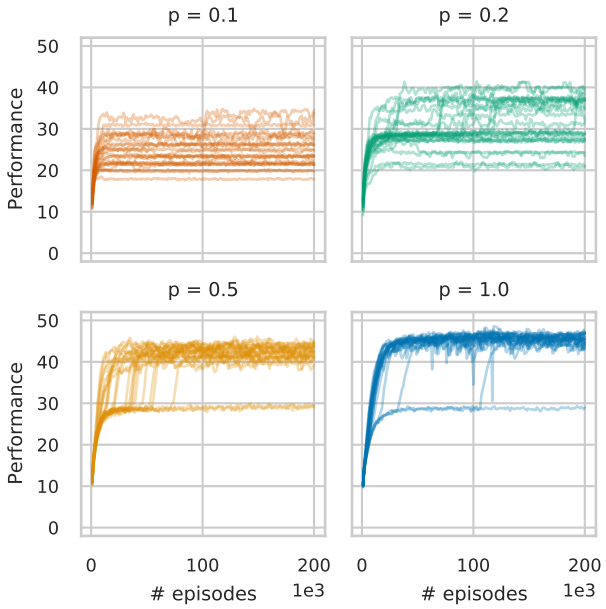

Figure 4.10: Split view of the different performances. For $p = 1.0$, in most of the simulations, the agent reaches a performance close to the optimal performance at the end of its learning. In a few simulations, the agent gets stuck at a plateau value. If the agent overcome this plateau value, it reaches quickly the close-to-optimal performance. For $p = 0.5$, in most of the simulation the agent's performance gets close to the optimal reward value. The plateau is reached more often than in the $p = 1.0$ condition. The plateau's equilibrium is even stronger for $p = 0.2$ and $p = 0.1$. In almost no simulation to no simulation at all the agent's performance escape from the plateau equilibrium to reach the close-to-optimum equilibrium.

The detailed analysis of the rewards through the episodes of the CMA-ES condition shows the same pattern as the PPO conditions. Most simulations converge towards a reward above 45, regardless of the value of p. A few rare simulations are stuck at a reward of 30. These stuck simulations are particularly rare compared to the PPO condition.

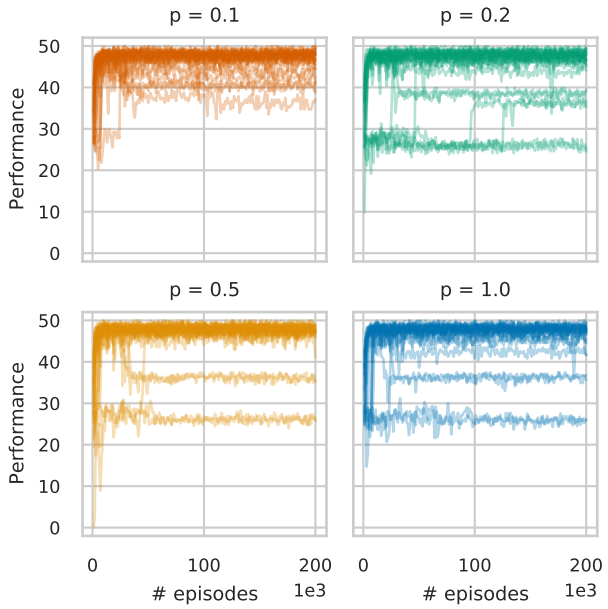

Figure 4.11: Split view of the different performances for different values of p. Regardless of the value p, in most of the simulations the agent gets a performance close to the optimal performance at the end of the learning. Like the PPO condition, some simulations reach a sub-optimal equilibrium around 30. Few of them manage to get out of this equilibrium.

4.6.2 Re-evaluation performance statistical score

We perform two-tailed Mann-Whitney's U-tests to compare the distributions of the performance for the PPO and CMA-ES agent for each probability p of meeting a x^+ partner. The table of the median performance of each learning algorithm for each p is reported in table 4.3 and the U-statistics and p-values of the tests are reported in table 4.4.

		Performance	
		Median	MAD
p	Algorithm		
0.1	CMA-ES	47.72	2.45
	PPO	24.11	3.37
0.2	CMA-ES	47.56	5.32
	PPO	32.38	5.15
0.5	CMA-ES	48.00	4.59
	PPO	42.55	2.57
1.0	CMA-ES	47.64	4.72
	PPO	45.59	1.85

Table 4.3: Median of the re-evaluations, 24 runs per condition

Condition p	U statistic	p-value
0.1	0	< 0.0001
0.2	59	< 0.0001
0.5	92	< 0.0001
1.0	172	1.724×10^{-2}

Table 4.4: Statistical results of the two-tailed Mann-Whitney U-test comparing the performance of the agents using PPO and CMA-ES in the re-evaluation setup. $n = 24$ for each condition and algorithm.

4.6.3 Timing

We measure the execution time (wall time) for both PPO and CMA-ES on a single CPU. Though, our CMA-ES implementation use far less overhead than RLlib's PPO implementation, which logs much more data, and that is built to be multi-core. This overhead may explain parts of difference in execution time. The total (both learning and episode evaluation) time for each condition and for each algorithm are plotted in figure 4.12. The total time and the total time divided by the number of iteration is reported in Table 4.5.

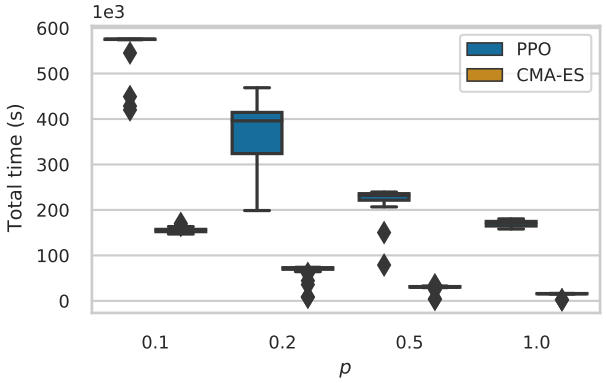

Figure 4.12: Total time used for the whole process between PPO and CMA-ES

		Total time (h)		time/iteration (ms)	
		mean	std	mean	std
p	Algorithm				
0.1	CMA-ES	43.31	1.46	1.21	0.02
	PPO	154.53	13.44	47.14	0.99
0.2	CMA-ES	17.68	5.40	1.17	0.07
	PPO	102.69	18.35	44.71	3.77
0.5.	CMA-ES	7.95	2.18	1.26	0.14
	PPO	61.01	9.82	41.80	1.31
1.0	CMA-ES	4.10	1.07	1.32	0.23
	PPO	47.11	1.87	44.54	1.75

Table 4.5: Table of computational wall time for PPO **(dual core and optimization)** and CMA-ES

Chapter 5

Conclusion

5.1 Summary

As presented in **Chapter 1** of this thesis, numerous requirements for partner choice have been uncovered. However, cooperative behaviours enforced by partner choice are seldom observed in nature. Most of the research so far has been focused on two agents interacting together. In these setups, agents can merely meet and start to interact. They do not need a resource to exploit together. This greatly simplifies the search for a new cooperation opportunity, as an agent only seeks for a partner. Many instances of cooperative interaction without an external resource can be found in nature. Social grooming in vervet monkeys, sexual markets or most inter-specific interactions are such examples. Despite that, there are also many tasks that require an external resource to exploit, be it alone or in a group. For instance, hunting requires a prey, or foraging require a patch of food; they all require a resource to exploit. In this context, individuals must find both a resource *and* a partner to cooperate with. Thus, both the abundance of partners *and* resources will impact the search time for an individual to find a cooperation opportunity. We predict that the scarcer the environment is, the harder it is for an individual to find a cooperation opportunity and thus that this will harm the partner choice efficiency. Thus, cooperation enforced by partner choice must be less robust.

In **Chapter 2**, We designed an individual-based model in which individuals must meet on resource patches to interact together. The resource patch must host a specific number of individuals to be exploited optimally. If there are too few or too many individuals on a resource patch, the payoff received by each agent is severely reduced. The number of individuals, the optimal number on a patch as the number of resource patches vary according to the

different experimental conditions.

We showed that when the number of resource patches in the environment is too low to host all the individuals, individuals play the selfish optimal investment value. They do not need to attract partners, so they do not need to invest more than teir selfish optimum. Though, when the number of patches allows all the individuals to interact at the optimal number of partners, partner choice mechanism develops and fixate in the population as an individual can easily find a patch of resource and wait for a partner if they cannot already find a patch with a satisfying partner. Furthermore, we investigate the impact of the optimal number of individual interacting. We show that when the optimal number is too high, that is numerous individuals must gather on the same resource patch to gather resources, they do not cooperate. Indeed, the negative marginal impact of a cheater in a big group of individuals goes unnoticed, and therefore the cheaters can exploit the cooperators.

In **Chapter 3**, we addressed the problem of learning cooperative strategies in swarm robotics. While the motivation differs from the one that drove the work presented in the previous Chapter, the questions raised turn out to be very similar as we now consider partner choice as a relevant mechanism to enforce learning optimal social behaviour in a group of independent learning agents. We were interested in heterogeneous robot swarms, in which each robotic agent optimises its individual gain. An important assumption is that we consider *individual learners*, which implies that the evolutionary algorithms cannot be used with a clonal approach (where all cooperators are genetically similar). For some tasks, the problem is that the optimal strategy requires to cooperate may be counter-selected in favour of a more stable but less efficient selfish strategy. To solve this problem, we exploit the mechanism of partner choice, which conditions of operation are learned by the agents. The striking difference with the work presented earlier in the thesis is that robotic agents actually have to learn to *move* in the environment. This implies both more complex interactions and control, as each individual can partially act on their ability to shorten the search for resources and partners.

We built a pseudo-realistic model where individuals must evolve both a navigation module and a partner choice module. Individuals navigate in the environment and when two individuals meet on a resource patch, they can decide to interact together or not, depending on their partner's and own investments. We studied the impact of the population density, which lower the seeking time, and the interaction time. We find consistent results with the previous aspatial models. When the population is dense enough, and the interaction time long enough, individuals act cooperatively and refuse

to interact with defectors. When the population is too sparse or when the interaction time is too low, individuals prefer to interact with the first partner they meet instead of spending time finding a better one.

It turns out that while partner choice can be very efficient at favouring the learning of socially efficient behaviours, the constraints are surprisingly strong as the population density as well as the interaction time must be considerable. On the one hand, these results extended those obtained in Chapter 2 by considering a spatial environment, and thus do contribute to a better understanding of partner choice in more realistic setups. On the other hand, we showed here that whenever individual learners such as robotic agents may cooperate, partner choice is beneficial with respect to the quality of cooperation if certain environmental conditions are met.

So far, we used evolutionary algorithms as a policy learning method for individual robotic agents, which raises the question as to whether other reinforcement learning methods may be beneficial when it comes to exploiting the possible benefits of partner choice in a setup where cooperation is possible.

In **Chapter 4** we have taken a step forward in this direction. In Chapters 2 and 3 we used fitness proportionate as a selection method for our evolutionary algorithms, which works at the level of the population, even though individuals are evaluated on an individual basis. This selection method is biologically realistic, but we implied that the population renewal process works in a centralised fashion, comparing performances of all individuals present in the environment. This constraint does not allow the design of robust and flexible robot swarms, where any learning algorithm should work in a decentralised fashion. In Chapter 4, we consider individual learning, where both performance assessment *and* learning occur solely at the level of the robot. Setting up an environment where different partners can be met, each with different strategies with respect to establishing cooperation, we first show that using individual learning algorithms does *not* change the expected results: both a state-of-the-art evolutionary learning method (CMA-ES for policy search) and a state-of-the-art reinforcement learning method (PPO, Proximal Policy Approximation) fared similar results. However, we also showed that not all learning methods are equal: PPO suffered greatly whenever meaningful interactions are sparse with respect to the total number of interactions, a problem to which CMAES is robust.

5.2 Discussion and Perspectives

Results described in this manuscript could lead to further studies. First of all, in Chapter 2, we allow individuals to be as many as they want to

be on the same resource patch. This capacity has only very rarely been explored in the models on partner choice. With the exception of Aktipis (2011), which allows as many agents as they wish to be on the same cell, all the models on partner choice are limited either to peer interactions or to the totality of individuals interacting together and sharing the gains afterwards (as in Barclay, 2011 for example). In our model on the influence of resource scarcity on partner choice behaviour, we allow agents to be as many as they want on a resource patch. We either set an optimal number of individuals on a resource for them to obtain the maximum gain, or we do not constrain this number of individuals. Very interestingly, we only obtain partner choice behaviour when the optimal number of individuals on a resource is two. As soon as we increase this number, for example to three or four, none of the agent populations manages to develop conditional cooperation. Although we expected that the ability of agents to make partner choice would diminish as we increased the number of partners required to carry out the task, we did not expect such a rapid collapse. Partner choice is less effective when the number of simultaneous partners for an agent increases because the greater the number of partners in an interaction, the easier it is to act as a free-rider, i.e. to cheat, without having a strong impact on the overall efficiency of the group. Thus, the cheater can obtain all the gains of an efficient group without paying the cost of cooperation.

This "dilution of responsibility" effect can be dramatic for group efficiency. Nevertheless, the study of this phenomenon could be extended beyond what we developed in Chapter 2. We could build a model focusing solely on the impact of group size on the mechanism of partner choice. We could analyse in more detail what parameters lead to the emergence of partner choice behaviour as a function of the number of individuals in a group. The main hypothesis is that the larger the group, the more complicated the partner choice behaviour is to acquire and the less efficient it is.

Partner choice may be more or less effective in large groups depending on several factors. Firstly, how the reward of the group is bound to the investment of each individual has a strong impact on the viability of partner choice. For example, if the reward is related to the group's minimum investment, then it is no longer possible for agents to cheat and to only have a marginal impact on group effectiveness. Their cheating action would have a detrimental impact on group efficiency and partner choice would be an effective mechanism counter selecting cheaters even in a large group. Furthermore, the information that agents have access to in deciding whether or not to join a group has a major impact on the effectiveness of the partner choice mechanism. Thus, if agents can only know the total investment of the group, but not the investment of each agent in the group, it is much harder

for agents to detect a free-rider.

Since partner choice only works effectively for small interaction groups, it would also be interesting to study how collective efficiency on a large set of individuals can be achieved using partner choice as a lever. Cooperation between individuals is the gateway to the realisation of complex tasks, possibly involving division of labour, with a hierarchy between tasks and sub-tasks. Studying whether cooperation by partner choice can lead to the emergence of specialised and structured populations is an exciting long-term objective to better understand the origins of societies between unrelated individuals.

In Chapter 3, we propose a pseudo-realistic robotic model in which agents move and interact together. Although the displacements are in a pseudo-realistic framework, the interactions between the agents are still a very abstract economic game. It would be interesting to find more concrete applications of the interactions between agents. For example, we could consider tasks such as moving objects or foraging resources back to a home base, as Ferrante et al. (2015) did. These new tasks, although they would not fundamentally change the results we have obtained, would allow us to prove that learning to cooperate by partner choice in robotic swarming can have real-world applications.

Finally, a particularly interesting element of partner choice was not explored in this manuscript. Partner choice can be implemented through a decision after an interaction (partner switching), but also through a decision before the interaction (partner *choice* stricto sensu). This ex-ante choice is possible thanks to the use of a memory or a reputation mechanism between the agents. In all our works, agents know whether they are with a cooperating partner only ex-post, after having interacted with them. Having *a priori* knowledge (Campennì and Schino, 2014 for example) allows a much shorter search time. Indeed, the meeting rate β presented in this thesis would be much higher if agents could distinguish cooperators from non-cooperators without having to interact with them each time. It would no longer be a question of meeting an individual, coordinating with him to go to a cooperation site, then identifying his behaviour, but simply seeing the partner and having an appreciation of its behaviour. If the partner is known to be good, then joining it at a site and cooperating with it is viable. This coordination time is finally included in the completion of the task itself, and therefore it would influence the parameter τ, the split rate of the agents. We have already tried to explore these reputational approaches in more complex models without success. Building a simple model to compare the behaviour of agents with and without memory and reputation mechanism would allow us to show the efficiency of these approaches. However, these mechanisms imply a more important cognitive processing on the part of the agents. Learning to manip-

ulate this information can be complex. This could be detrimental to learning and agents may not be able to find the partner choice behaviour because of this complex information to interpret.

Bibliography

Aktipis, C. A. (2004). Know when to walk away: Contingent movement and the evolution of cooperation. *Journal of Theoretical Biology*, *231*(2), 249–260.

Aktipis, C. A. (2011). Is cooperation viable in mobile organisms? Simple Walk Away rule favors the evolution of cooperation in groups. *Evolution and Human Behavior*, *32*(4), 263–276.

Alvard, M. S., & Nolin, D. A. (2002). Rousseaus Whale Hunt? *Current Anthropology*, *43*(4), 533–559.

Amato, C., Konidaris, G., Cruz, G., Maynor, C. A., How, J. P., & Kaelbling, L. P. (2015). Planning for decentralized control of multiple robots under uncertainty, In *Proceedings - ieee international conference on robotics and automation*. IEEE.

Andersson, M., & Simmons, L. W. (2006). Sexual selection and mate choice. *Trends in Ecology and Evolution*, *21*(6), 296–302.

André, J. B. (2014). Mechanistic constraints and the unlikely evolution of reciprocal cooperation. *Journal of Evolutionary Biology*, *27*(4), 784–795.

André, J. B. (2015). Contingency in the evolutionary emergence of reciprocal cooperation. *American Naturalist*, *185*(3), 303–316.

André, J. B., & Baumard, N. (2011a). Social opportunities and the evolution of fairness. *Journal of Theoretical Biology*, *289*(1), 128–135.

André, J. B., & Baumard, N. (2011b). The evolution of fairness in a biological market. *Evolution*, *65*(5), 1447–1456.

André, J. B., & Day, T. (2007). Perfect reciprocity is the only evolutionarily stable strategy in the continuous iterated prisoner's dilemma. *Journal of Theoretical Biology*, *247*(1), 11–22.

André, J. B., & Nolfi, S. (2016). Evolutionary robotics simulations help explain why reciprocity is rare in nature. *Scientific Reports*, *6*(1), 32785.

Axelrod, R., & Hamilton, W. D. (1981). The evolution of cooperation. *Science*, *212*(4489), 1390–1396.

Baldassarre, G., Nolfi, S., & Parisi, D. (2003). Evolving mobile robots able to display collective behaviors. *Artificial Life*, *9*(3), 255–267.

Baray, C. (1997). Evolving cooperation via communication in homogeneous multi-agent systems, In *Proceedings - intelligent information systems, iis 1997*, Institute of Electrical; Electronics Engineers Inc.

Barclay, P. (2011). Competitive helping increases with the size of biological markets and invades defection. *Journal of Theoretical Biology*, *281*(1), 47–55.

Barclay, P. (2013). Strategies for cooperation in biological markets, especially for humans. *Evolution and Human Behavior*, *34*(3), 164–175.

Barclay, P. (2016). Biological markets and the effects of partner choice on cooperation and friendship. *Current Opinion in Psychology*, *7*, 33–38.

Barclay, P., & Willer, R. (2007). Partner choice creates competitive altruism in humans. *Proceedings of the Royal Society B: Biological Sciences*, *274*(1610), 749–753.

Baumard, N., André, J. B., & Sperber, D. (2013). A mutualistic approach to morality: The evolution of fairness by partner choice. *Behavioral and Brain Sciences*, *36*(1), 59–78.

Bayindir, L. (2016). A review of swarm robotics tasks. *Neurocomputing*, *172*, 292–321.

Beni, G. (2005). From swarm intelligence to swarm robotics. *Lecture Notes in Computer Science*, *3342*, 1–9.

Bernard, A., André, J. B., & Bredeche, N. (2016). To Cooperate or Not to Cooperate: Why Behavioural Mechanisms Matter. *PLoS Computational Biology*, *12*(5), 1–14.

Bernard, A., Bredeche, N., & André, J. (2020). Indirect genetic effects allow escape from the inefficient equilibrium in a coordination game. *Evolution Letters*, *4*(3), 257–265.

Bhatnagar, S., Borkar, V. S., & Akarapu, M. (2006). A simulation-based algorithm for ergodic control of markov chains conditioned on rare events. *Journal of Machine Learning Research*, *7*(Oct), 1937–1962.

Bongard, J. C., Bongard, J. C., & Paul, C. (2000). Investigating Morphological Symmetry and Locomotive Efficiency using Virtual Embodied Evolution, In *From animals to animats 6*. The MIT Press.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, *7*(1), 1–41.

Bredeche, N., Haasdijk, E., & Prieto, A. (2018). Embodied evolution in collective robotics: A review. *Frontiers Robotics AI*, *5*(FEB), 12.

Bredeche, N., Montanier, J.-M., Weel, B., & Haasdijk, E. (2013). Roborobo! a Fast Robot Simulator for Swarm and Collective Robotics, 1–2.

Bryant, B. D., & Miikkulainen, R. (2003). *Neuroevolution for Adaptive Teams* (tech. rep.).

Bshary, R., & Grutter, A. S. (2002). Experimental evidence that partner choice is a driving force in the payoff distribution among cooperators or mutualists: The cleaner fish case. *Ecology Letters*, *5*(1), 130–136.

Bshary, R., & Grutter, A. S. (2005). Punishment and partner switching cause cooperative behaviour in a cleaning mutualism. *Biology Letters*, *1*(4), 396–399.

Bshary, R., & Grutter, A. S. (2006). Image scoring and cooperation in a cleaner fish mutualism. *Nature*, *441*(7096), 975–978.

Bshary, R., & Noe, R. (2003). Biological Markets The Ubiquitous Influence of Partner Choice on the Dynamics of Cleaner. *Genetic and Cultural Evolution of Cooperation*, 167–184.

Bull, J. J., & Rice, W. R. (1991). Distinguishing mechanisms for the evolution of co-operation. *Journal of Theoretical Biology*, *149*(1), 63–74.

Bullinger, A. F., Melis, A. P., & Tomasello, M. (2011). Chimpanzees, Pan troglodytes, prefer individual over collaborative strategies towards goals. *Animal Behaviour*, *82*(5), 1135–1141.

Camazine, S., Deneubourg, J.-L., Franks, N., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2001). *Self-organization in biological systems*. Princeton University Press.

Campennì, M., & Schino, G. (2014). Partner choice promotes cooperation: The two faces of testing with agent-based models. *Journal of Theoretical Biology*, *344*, 49–55.

Carter, G. (2014). The Reciprocity Controversy. *Animal Behavior and Cognition*, *1*(3), 368.

Chade, H., Eeckhout, J., & Smith, L. (2017). Sorting through Search and matching models in economics. *Journal of Economic Literature*, *55*(2), 493–544.

Ciosek, K., & Whiteson, S. (2017). Offer: Off-environment reinforcement learning, In *Proceedings of the aaai conference on artificial intelligence*.

Clutton-Brock, T. (2009). Cooperation between non-kin in animal societies. *Nature*, *462*(7269), 51–57.

Davies, N. B. (2012). *An introduction to behavioural ecology*. Wiley-Blackwell.

Debove, S., André, J. B., & Baumard, N. (2015). Partner choice creates fairness in humans. *Proceedings of the Royal Society B: Biological Sciences*, *282*(1808), 1–7.

Debove, S., Baumard, N., & André, J. B. (2015). Evolution of equal division among unequal partners. *Evolution*, *69*(2), 561–569.

Debove, S., Baumard, N., & André, J. B. (2017). On the evolutionary origins of equity. *PLoS ONE*, *12*(3), 5–7.

Dittami, J. (2001). *Economics in Nature* (R. Noë, J. A. R. A. M. Van Hooff, & P. Hammerstein, Eds.; Vol. 109). Cambridge University Press.

Doncieux, S. (2015). Representational redescription: the next challenge? *AMD Newsletter*, *12*(1), 16–17.

Doncieux, S., Bredeche, N., Mouret, J. B., & (Gusz) Eiben, A. E. (2015). Evolutionary robotics: What, why, and where to. *Frontiers Robotics AI*, *2*(MAR), 1–18.

dos Santos, M., & West, S. A. (2018). The coevolution of cooperation and cognition in humans. *Proceedings of the Royal Society B: Biological Sciences*, *285*(1879).

Dunbar, R. I., & Shultz, S. (2007). Evolution in the social brain. *Science*, *317*(5843), 1344–1347.

Eshel, I., & Cavalli-Sforza, L. L. (1982). Assortment of encounters and evolution of cooperativeness. *Proceedings of the National Academy of Sciences of the United States of America*, *79*(4 I), 1331–1335.

Ferrante, E., Turgut, A. E., Duéñez-Guzmán, E., Dorigo, M., & Wenseleers, T. (2015). Evolution of Self-Organized Task Specialization in Robot Swarms (O. Sporns, Ed.). *PLoS Computational Biology*, *11*(8), e1004273.

Frank, J., Mannor, S., & Precup, D. (2008). Reinforcement learning in the presence of rare events, In *Proceedings of the 25th international conference on machine learning*.

Fruteau, C., Voelkl, B., Van Damme, E., & Noë, R. (2009). Supply and demand determine the market value of food providers in wild vervet monkeys. *Proceedings of the National Academy of Sciences of the United States of America*, *106*(29), 12007–12012.

Fudenberg, D. (1998). *The theory of learning in games* (Vol. 36). Cambridge, MA, MIT Press.

Geoffroy, F., Baumard, N., & André, J.-B. (2018). Why cooperation is not running away. *bioRxiv*, 316117.

Guide, T., & S, T. (2003). *Natural Computing Series*. Berlin, Heidelberg, Springer Berlin Heidelberg.

Haasdijk, E., Bredeche, N., & Eiben, A. E. (2014). Combining environment-driven adaptation and task-driven optimisation in evolutionary robotics (E. Vasilaki, Ed.). *PLoS ONE*, *9*(6), e98466.

Hamann, H. (2018). *Swarm Robotics: A Formal Approach*. Springer.

Hamilton, W. D. (1964). The genetical evolution of social behaviour. I. *Journal of Theoretical Biology*, *7*(1), 1–16.

Hammerstein, P., & Noë, R. (2016). Biological trade and markets. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *371*(1687), 20150101.

Hansen, N., & Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, *9*(2), 159–195.

Hansen, N., Akimoto, Y., & Baudis, P. (2020). *Cma-es/pycma: R3.0.3* (Version r3.0.3). Zenodo. https://doi.org/10.5281/zenodo.3764210

Hauert, S., Mitri, S., Keller, L., & Floreano, D. (2019). Evolving Cooperation: From Biology to Engineering, In *The horizons of evolutionary robotics*. MIT Press.

Heinerman, J., Drupsteen, D., & Eiben, A. E. (2015). Three-fold adaptivity in groups of robots: The effect of social learning (S. Silva, Ed.). In S. Silva (Ed.), *Gecco 2015 - proceedings of the 2015 genetic and evolutionary computation conference*, ACM.

Henrich, J., & McElreath, R. (2003). The evolution of cultural evolution. *Evolutionary Anthropology: Issues, News, and Reviews*, *12*(3), 123–135.

Jaderberg, M., Mnih, V., Czarnecki, W. M., Schaul, T., Leibo, J. Z., Silver, D., & Kavukcuoglu, K. (2017). Reinforcement learning with unsupervised auxiliary tasks, In *5th international conference on learning representations, ICLR 2017, toulon, france, april 24-26, 2017, conference track proceedings*.

Johnstone, R. A., & Bshary, R. (2008). Mutualism, market effects and partner control. *Journal of Evolutionary Biology*, *21*(3), 879–888.

Kaplan, H., Hill, K., Lancaster, J., & Hurtado, A. M. (2000). A theory of human life history evolution: Diet, intelligence, and longevity. *Evolutionary Anthropology: Issues, News, and Reviews*, *9*(4), 156–185.

Kawecki, T. J., Lenski, R. E., Ebert, D., Hollis, B., Olivieri, I., & Whitlock, M. C. (2012). Experimental evolution. *Trends in Ecology and Evolution*, *27*(10), 547–560.

Konidaris, G., & Barto, A. (2006). Autonomous shaping: Knowledge transfer in reinforcement learning, In *Proceedings of the 23rd international conference on machine learning*.

Krams, I., Krama, T., Igaune, K., & Mänd, R. (2008). Experimental evidence of reciprocal altruism in the pied flycatcher. *Behavioral Ecology and Sociobiology*, *62*(4), 599–605.

Lenski, R. E., Mongold, J. A., Sniegowski, P. D., Travisano, M., Vasi, F., Gerrish, P. J., & Schmidt, T. M. (1998). *Evolution of competitive fitness in experimental populations of E. coli: What makes one genotype a better competitor than another?* (Tech. rep.).

Luke, S., Hohn, C., Farris, J., Jackson, G., & Hendler, J. (1998). Co-evolving Soccer Softbot team coordination with genetic programming. Springer, Berlin, Heidelberg.

Marden, J. R., & Wierman, A. (2008). Distributed welfare games with applications to sensor coverage, In *Proceedings of the ieee conference on decision and control*.

Mataric, M. J. (1992). Integration of Representation Into Goal-Driven Behavior-Based Robots. *IEEE Transactions on Robotics and Automation*, *8*(3), 304–312.

Maynard Smith, J. (1974). The theory of games and the evolution of animal conflicts. *Journal of Theoretical Biology*, *47*(1), 209–221.

McNamara, J. M., Barta, Z., Fromhage, L., & Houston, A. I. (2008). The coevolution of choosiness and cooperation. *Nature*, *451*(7175), 189–192.

McNamara, J. M., & Leimar, O. (2010). Variation and the response to variation as a basis for successful cooperation. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *365*(1553), 2627–2633.

Melis, A. P., Hare, B., & Tomasello, M. (2008). Do chimpanzees reciprocate received favours? *Animal Behaviour*, *76*(3), 951–962.

Melis, A. P., Schneider, A. C., & Tomasello, M. (2011). Chimpanzees, Pan troglodytes, share food in the same way after collaborative and individual food acquisition. *Animal Behaviour*, *82*(3), 485–493.

Mitri, S., Wischmann, S., Floreano, D., & Keller, L. (2013). Using robots to understand social behaviour. *Biological Reviews*, *88*(1), 31–39.

Montanier, J. M., Carrignon, S., & Bredeche, N. (2016). Behavioral specialization in embodied evolutionary robotics: Why so difficult? *Frontiers Robotics AI*, *3*(JUL), 1.

Murray, J. D. (2002). *Mathematical biology*. New York, Springer.

Noë, R. (2006). Cooperation experiments: Coordination through communication versus acting apart together. Academic Press.

Noë, R., & Hammerstein, P. (1994). Biological markets: supply and demand determine the effect of partner choice in cooperation, mutualism and mating. *Behavioral Ecology and Sociobiology*, *35*(1), 1–11.

Nolfi, S., Floreano, D., & Floreano, D. D. (2000). *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*.

Nowak, M. A. (2006). Five rules for the evolution of cooperation. *Science*, *314*(5805), 1560–1563.

Packer, C., Scheel, D., & Pusey, A. E. (1990). Why lions form groups: food is not enough. *American Naturalist*, *136*(1), 1–19.

Packer, C. (2014). 19. The Ecology of Sociality in Felids (D. I. Rubenstein & R. W. Wrangham, Eds.). In D. I. Rubenstein & R. W. Wrang-

ham (Eds.), *Ecological aspects of social evolution*. Princeton, Princeton University Press.

Packer, G., & Ruttan, L. (1988). The evolution of cooperative hunting. *American Naturalist*, *132*(2), 159–198.

Polak, I., & Abdou, J. (2014). *Reducing Evolutionary Stability to Pure Strategies in Positive Semidefinite Games* (tech. rep.).

Quinn, M., Smith, L., Mayley, G., & Husbands, P. (2002). Evolving Formation Movement for a Homogeneous Multi-Robot System : Teamwork and Role-Allocation with Real Robots . Cognitive Science Research Papers Evolving Formation Movement for a Homogeneous Multi-Robot System : Teamwork and Role-Allocation with Real. *Robotics*, (May).

Raihani, N. J., & Bshary, R. (2011). Resolving the iterated prisoner's dilemma: Theory and reality. *Journal of Evolutionary Biology*, *24*(8), 1628–1639.

Railsback, S. F., & Grimm, V. (2019). *Agent-based and individual-based modeling: a practical introduction*. Princeton university press.

Riedmiller, M., Hafner, R., Lampe, T., Neunert, M., Degrave, J., Wiele, T., Mnih, V., Heess, N., & Springenberg, J. T. (2018). Learning by playing solving sparse reward tasks from scratch, In *International conference on machine learning*.

Robson, A. J. (1990). Efficiency in evolutionary games: Darwin, nash and the secret handshake. *Journal of Theoretical Biology*, *144*(3), 379–396.

Sachs, J. L., Mueller, U. G., Wilcox, T. P., & Bull, J. J. (2004). The evolution of cooperation. *Quarterly Review of Biology*, *79*(2), 135–160.

Salimans, T., Ho, J., Chen, X., Sidor, S., & Sutskever, I. (2017). Evolution Strategies as a Scalable Alternative to Reinforcement Learning, 1–13.

Scheel, D., & Packer, C. (1991). Group hunting behaviour of lions: a search for cooperation. *Animal Behaviour*, *41*(4), 697–709.

Schino, G. (2007). Grooming and agonistic support: A meta-analysis of primate reciprocal altruism. *Behavioral Ecology*, *18*(1), 115–120.

Schino, G., & Aureli, F. (2008). Grooming reciprocation among female primates: A meta-analysis. *Biology Letters*, *4*(1), 9–11.

Schino, G., & Aureli, F. (2017). Reciprocity in group-living animals: Partner control versus partner choice. *Biological Reviews*, *92*(2), 665–672.

Schroeder, D. A., Graziano, W. G., Barclay, P., & Van Vugt, M. (2014). The Evolutionary Psychology of Human Prosociality. *The Oxford Handbook of Prosocial Behavior*, 37–60.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms.

Schulze, J., Müller, B., Groeneveld, J., & Grimm, V. (2017). Agent-based modelling of social-ecological systems: Achievements, challenges, and a way forward. *Jasss*, *20*(2).

Shoham, Y. (2009). *Multiagent systems : algorithmic, game-theoretic, and logical foundations*. Cambridge New York, Cambridge University Press.

Simms, E. L., & Lee Taylor, D. (2002). Partner choice in nitrogen-fixation mutualisms of legumes and rhizobia. *Integrative and Comparative Biology*, *42*(2), 369–380.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Taylor, M. E., Whiteson, S., & Stone, P. (2006). Comparing evolutionary and temporal difference methods in a reinforcement learning domain. *GECCO 2006 - Genetic and Evolutionary Computation Conference*, *2*, 1321–1328.

Trianni, V. (2008a). *Evolutionary swarm robotics : evolving self-organising behaviours in groups of autonomous robots* (Vol. 108). Springer.

Trianni, V. (2008b). From solitary to collective behaviours: Decision making and cooperation. *Studies in Computational Intelligence*, *108*, 161–170.

Trianni, V. (2014). Evolutionary Robotics: Model or Design? *Frontiers in Robotics and AI*, *1*, 13.

Trianni, V., & Dorigo, M. (2006). Self-organisation and communication in groups of simulated and physical robots. *Biological Cybernetics*, *95*(3), 213–231.

Trivers, R. L. (1971). The Evolution of Reciprocal Altruism. *The Quarterly Review of Biology*, *46*(1), 35–57.

Tuci, E., & Trianni, V. (2014). On the evolution of homogeneous two-robot teams: clonal versus aclonal approaches. *Neural Computing and Applications*, *25*(5), 1063–1076.

Verbancsics, P., & Stanley, K. O. (2010). Evolving static representations for task transfer. *Journal of Machine Learning Research*, *11*, 1737–1769.

Waibel, M., Floreano, D., & Keller, L. (2011). A quantitative test of Hamilton's rule for the evolution of altruism (N. H. Barton, Ed.). *PLoS Biology*, *9*(5), 1–7.

Waibel, M., Keller, L., & Floreano, D. (2009). Genetic team composition and level of selection in the evolution of cooperation. *IEEE Transactions on Evolutionary Computation*, *13*(3), 648–660.

Watson, R. A., Ficici, S. G., & Pollack, J. B. (2002). Embodied Evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, *39*(1), 1–18.

Watson, R. A., Ficiei, S. G., & Pollack, J. B. (1999). Embodied evolution: Embodying an evolutionary algorithm in a population of robots. *Pro-*

ceedings of the 1999 Congress on Evolutionary Computation, CEC 1999, *1*, 335–342.

West, S. A., Griffin, A. S., & Gardner, A. (2007a). Social semantics: Altruism, cooperation, mutualism, strong reciprocity and group selection. *Journal of Evolutionary Biology*, *20*(2), 415–432.

West, S. A., Griffin, A. S., & Gardner, A. (2007b). Evolutionary Explanations for Cooperation. *Current Biology*, *17*(16), 661–672.

Wilkinson, G. S., Carter, G. G., Bohn, K. M., & Adams, D. M. (2016). Non-kin cooperation in bats. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *371*(1687).

Wischmann, S., Floreano, D., & Keller, L. (2012). Historical contingency affects signaling strategies and competitive abilities in evolving populations of simulated robots. *Proceedings of the National Academy of Sciences of the United States of America*, *109*(3), 864–868.

Wolpert, D. H., & Nasagov, P. (2000). *Optimal Wonderful Life Utility Functions in Multi-Agent Systems Kagan Tumer* (tech. rep.).

Zahavi, A. (1975). Mate selection-A selection for a handicap. *Journal of Theoretical Biology*, *53*(1), 205–214.