

Evolution of division of labor in artificial societies

THÈSE N° 5687 (2013)

PRÉSENTÉE LE 22 MARS 2013

À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR
LABORATOIRE DE SYSTÈMES INTELLIGENTS
PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Paweł LICHOCKI

acceptée sur proposition du jury:

Prof. A. Billard, présidente du jury
Prof. D. Floreano, Prof. L. Keller, directeurs de thèse
Prof. M.-O. Hongler, rapporteur
Prof. L. Lehmann, rapporteur
Prof. F. Weissing, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2013

Acknowledgements

I received much support in preparing and writing this thesis. Above all, I am in debt to Dario Floreano and Laurent Keller, my two supervisors for accepting me in their labs. They granted me an exceptional freedom to pursue research subjects of my choosing, and repeatedly aided me with their vast expertise. I also owe a lot to all past and current members of the Laboratory of Intelligent Systems who created the exciting atmosphere that made my days in the lab so enjoyable and productive. Thank you all for many stimulating discussions, seminar meetings and valuable comments on my work. Special thanks go to Steffen Wischman with whom I shared the office, and to Danesh Tarapore, Markus Waibel, Sara Mitri, Nicolas Alcala, Andrea Maesani, Pavan Ramdya, Michał Dobrzyński, Maja Varga, Jürg Germann, Pradeep Fernando, and Thomas Shafter. Last, but definitely not least, I am grateful to my wife Barbara Piasecka for bearing the endless, boring conversations about my work that have let me crystalize many of the ideas presented here.

Lausanne, 22 February 2012

Paweł Lichocki

Abstract

Natural and artificial societies often divide the workload between specialized members. For example, an ant worker may preferentially perform one of many tasks such as brood rearing, foraging and nest maintenance. A robot from a rescue team may specialize in search, obstacle removal, or transportation. Such division of labor is considered crucial for efficient operation of multi-agent systems and has been studied from two perspectives. First, scientists address the "how" question seeking for mechanical explanations of division of labor. The focus has been put on behavioral and environmental factors and on task allocation algorithms leading to specialization. Second, scientists address the "why" question uncovering the origins of division of labor. The focus has been put on evolutionary pressures and optimization procedures giving rise to specialization. Studies have usually addressed one of these two questions in isolation, but for a full understanding of division of labor the explanation of the origins of specific mechanisms is necessary. Here, we rise to this challenge and study three major transitions related to division of labor. By means of theoretical analyses and evolutionary simulations, we construct a pathway from the occurrence of cooperation, through fixed castes, up to dynamic task allocation.

First, we study conditions favoring the evolution of cooperation, as it opens the doors for the potentially following specialization. We demonstrate that these conditions are sensitive to the mechanisms of intra-specific selection (or "selection methods"). Next, we take an engineering perspective and we study division of labor at the genetic level in teams of artificial agents. We devise efficient algorithms to evolve fixed assignments of agents to castes (or "team compositions"). To this end, we propose a novel technique that exchanges agents between teams, which greatly eases the search for the optimal composition. Finally, we take a biological perspective and we study division of labor at the behavioral level in simulated ant colonies. We quantify the efficiency of task allocation algorithms, which have been used to explain specialization in social insects. We show that these algorithms fail to induce precise reallocation of the workforce in response to changes in the environment. We overcome this issue by modeling task allocation with artificial neural networks, which lead to near optimal colony performance.

Abstract

Overall, this work contributes both to biology and to engineering. We shed light on the evolution of cooperation and division of labor in social insects, and we show how to efficiently optimize teams of artificial agents. We resolve the encountered methodological issues and demonstrate the power of evolutionary simulations to address biological questions and to tackle engineering problems.

Keywords: artificial evolution, multi-agent systems, cooperation, division of labor, specialization, team composition, task allocation, response thresholds, artificial neural networks, simulations, evolutionary computation, selection method, crossover

Résumé

Les sociétés naturelles et artificielles divisent souvent la charge de travail entre des membres spécialisés. Par exemple, une fourmi ouvrière peut préférentiellement effectuer l'une de nombreuses tâches telles que l'élevage du couvain, la recherche de nourriture ou l'entretien du nid. Un robot d'une équipe de secours peut se spécialiser dans la recherche, la suppression d'obstacles, ou le transport. La division du travail est considérée comme cruciale pour le fonctionnement efficace des systèmes multi-agents et a été étudié sous deux angles. Tout d'abord, les scientifiques abordent la question du «comment» et cherchent des explications mécaniques à la division du travail. L'accent est mis sur les facteurs comportementaux et environnementaux et sur les algorithmes d'allocation des tâches qui conduisent à la spécialisation. Deuxièmement, les scientifiques abordent la question du «pourquoi» et cherchent les origines de la division du travail. L'accent est mis sur les pressions évolutives et sur les procédures d'optimisation qui provoquent la spécialisation. Les études ont généralement traité ces deux questions séparément, mais pour une compréhension complète de la division du travail l'explication de l'origine des mécanismes spécifiques est nécessaire. Ici, nous relevons ce défi et nous étudions trois transitions majeures relatives à la division du travail. Par des analyses théoriques et des simulations évolutives, nous établissons un scénario menant de l'apparition de la coopération et de castes fixes jusqu'à l'allocation dynamique de tâches.

Tout d'abord, nous étudions les conditions qui favorisent l'évolution de la coopération, car cela ouvre la porte à une spécialisation plus poussée. Nous démontrons que ces conditions sont sensibles aux mécanismes de sélection intra-spécifique (ou «méthodes de sélection»). Ensuite, nous prenons un point de vue technique et nous étudions la division du travail au niveau génétique au sein d'équipes d'agents artificiels. Nous concevons des algorithmes efficaces permettant de faire évoluer les assignations fixes d'agents à des castes (ou «compositions des équipes»). À cette fin, nous proposons une nouvelle technique d'échange d'agents entre les équipes, ce qui facilite grandement la recherche de la composition optimale. Enfin, nous nous plaçons d'un point de vue biologique et nous étudions la division du travail au niveau comportemental dans les colonies de fourmis simulées. Nous quantifions l'efficacité des algorithmes d'allocation

Résumé

des tâches, qui ont été utilisés pour expliquer la spécialisation chez les insectes sociaux. Nous montrons que ces algorithmes n'induisent pas de répartition précise des ouvrières en réponse aux changements environnementaux. Nous remédions à ce problème par une modélisation de la répartition des tâches avec les réseaux de neurones artificiels, grâce à laquelle les colonies peuvent atteindre une performance quasi optimale.

Dans l'ensemble, nous contribuons à la biologie et à l'ingénierie. Nous éclairons l'évolution de la coopération et de la division du travail chez les insectes sociaux, et nous montrons comment optimiser efficacement des équipes d'agents artificiels. Nous résolvons les problèmes méthodologiques rencontrés et démontrons la puissance de simulations évolutives pour répondre à des questions biologiques et pour s'attaquer à des problèmes d'ingénierie.

Mots clés : évolution artificielle, systèmes multi-agents, coopération, division du travail, spécialisation, composition des équipes, allocation des tâches, seuil de réponse, réseaux neuronaux artificiels, simulations, algorithmes évolutionnaires, méthode de sélection, enjambement

Contents

Acknowledgements	iii
Abstract (English/Français)	v
List of figures	xi
List of tables	xv
Introduction	1
1 Selection methods regulate evolution of cooperation in artificial evolution	15
1.1 Disclosure	16
1.2 Abstract	16
1.3 Introduction	16
1.4 Methods	19
1.4.1 Selection methods	19
1.4.2 Model of cooperation	19
1.4.3 Evolutionary experiments	20
1.4.4 Genetic architecture, selection and reproduction	20
1.4.5 Statistical analysis	20
1.5 Results	21
1.5.1 Formal analyses	21
1.5.2 Simulations	22
1.6 Discussion	28
1.7 Supplementary materials	31
1.7.1 Mathematical model of cooperation in populations of related individuals	31
1.7.2 Evolution of cooperation under proportionate selection method	33
1.7.3 Evolution of cooperation under rank selection method	34
1.7.4 Evolution of cooperation under truncation selection methods	34

Contents

1.7.5	Evolution of cooperation under tournament selection method . . .	38
1.7.6	Effects of random drift	40
2	Evolving team compositions by agent swapping	45
2.1	Disclosure	46
2.2	Abstract	46
2.3	Introduction	47
2.3.1	Background	51
2.4	Methods	53
2.4.1	Restricted and free agent swapping	53
2.4.2	Problem formulation	55
2.4.3	Evolutionary experiments	56
2.4.4	Genetic architecture, selection and reproduction	56
2.4.5	Statistical analysis	57
2.5	Results	58
2.5.1	Restricted and free agent swapping	58
2.5.2	Restricted and free agent swapping - disparities in the initial pop- ulation	61
2.5.3	Combining free and restricted agent swapping	62
2.6	Discussion	64
2.7	Supplementary materials	68
2.7.1	1-point and 2-point crossover	68
2.7.2	Effects of changing agent swapping probability	71
2.7.3	Effects of mutation	72
2.7.4	Effects of selection pressure	74
2.7.5	Best team performance in the population	76
2.7.6	Alternative methods of combining team crossover operators	77
2.7.7	Other operators altering team composition	79
3	Neural networks as mechanisms to regulate division of labor	81
3.1	Disclosure	82
3.2	Abstract	82
3.3	Introduction	83
3.4	Methods	84
3.4.1	Task allocation mechanism	84
3.4.2	Colony tasks	86
3.4.3	Genetic architecture, selection and reproduction	88
3.4.4	Statistical analysis	89

3.5	Results	89
3.5.1	Formal analyses	89
3.5.2	Simulations	90
3.6	Discussion	94
3.7	Supplementary materials	98
3.7.1	Mathematical model of task allocation	98
3.7.2	Behavioral flexibility in response thresholds models	99
3.7.3	Analyses of other response thresholds models	101
3.7.4	Effects of population size and mutation	103
 Conclusions		 107
 A Association between performance and fitness in evolutionary simulations		 111
 B Application of team crossover operators in the evolution of decentralized controllers for task allocation		 115
 C Formal definition of the task allocation simulation		 119
 Bibliography		 142
 Curriculum Vitae		 143

List of Figures

1.1	Mean cooperation level evolved under different selection methods, the average over all conditions	23
1.2	Mean cooperation level evolved under different selection methods, the average over conditions with the same relatedness level and the average over conditions with the same benefit of cooperation	24
1.3	Mean cooperation level evolved under different selection methods for every condition	25
1.4	Mean cooperation level evolved under different selection methods for conditions with fixed benefit of cooperation	26
1.5	Mean stability of the cooperation level evolved under different selection methods	27
2.1	Two approaches in the evolution of heterogenous teams	48
2.2	Crossover operators acting on teams of agents	50
2.3	Restricted and free agent swapping	54
2.4	Mean performance of teams evolved with restricted and with free agent swapping	59
2.5	Mean variation between the teams evolved with restricted and with free agent swapping	59
2.6	Structure of teams evolved with restricted and with free agent swapping	60
2.7	Mean performance of teams evolved with restricted and with free agent swapping without disparities in the initial population	62
2.8	Mean performance of teams evolved with restricted, with free, and with mixed free/restricted agent swapping	63
2.9	Mean variation between the teams evolved with mixed free/restricted agent swapping	64
2.10	Mean performance of teams evolved with 1-point and with 2-point crossover	69
2.11	Mean variation between teams evolved with 1-point and with 2-point crossover	69

List of Figures

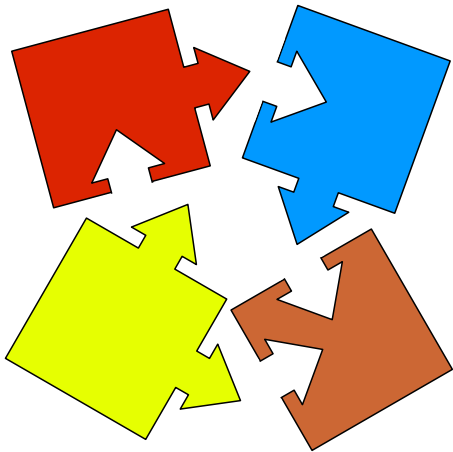
2.12	Structure of teams evolved with 1-point and with 2-point crossover	70
2.13	Mean performance of teams evolved with restricted and with free agent swapping, using low probability of agent swapping	71
2.14	Mean variation between teams evolved with restricted and with free agent swapping, using low probability of agent swapping	71
2.15	Structure of teams evolved with restricted and with free agent swapping, using low probability of agent swapping	72
2.16	Mean performance of teams evolved with restricted and with free agent swapping, using mutation	73
2.17	Mean variation between teams evolved with restricted and with free agent swapping, using mutation	73
2.18	Structure of teams evolved with restricted and with free agent swapping, using mutation	74
2.19	Mean performance of teams evolved with restricted and with free agent swapping, using high selection pressure	75
2.20	Mean variation between teams evolved with restricted and with free agent swapping, using high selection pressure	75
2.21	Structure of teams evolved with restricted and with free agent swapping, using high selection pressure	76
2.22	Best performance of teams evolved with restricted and with free agent swapping	77
2.23	Alternative ways of combining free and restricted agent swapping, mean performance of and mean variation between teams	78
3.1	Task allocation mechanisms and the neuronal formalism	85
3.2	Mean performance in task allocation of colonies evolved with deterministic, probabilistic and extended response threshold model	91
3.3	Distribution of workers to tasks in colonies evolved with deterministic, probabilistic and extended response threshold model, the average over colony lifespan	91
3.4	Distribution of workers to tasks in colonies evolved with deterministic, probabilistic and extended response threshold model, in a function of time	92
3.5	Proportion of workers switching between tasks in colonies evolved with deterministic, probabilistic and extended response threshold model, the average over colony lifespan	93
3.6	Efficiency in the regulatory task in colonies evolved with deterministic, probabilistic and extended response threshold model	93

3.7	Mean proportion of colonies evolved evolved with deterministic, probabilistic and extended response threshold model that successfully perform regulatory task in a function of time	94
3.8	Mean performance in task allocation of colonies evolved with alternative versions of deterministic and probabilistic response threshold models . . .	101
3.9	Distribution of workers to tasks in colonies evolved with alternative versions of deterministic and probabilistic response threshold models, in a function of lifetime	102
3.10	Mean performance in task allocation of colonies evolved with deterministic, probabilistic and extended response threshold model, using different population sizes and different mutations	104
3.11	Distribution of workers to tasks in colonies evolved with alternative versions of deterministic and probabilistic response threshold models, in a function of lifetime, using different population sizes and different mutations	105
A.1	Association between fitness and performance under proportionate, rank, truncation-proportionate, truncation-uniform, and tournament selection methods	112
B.1	Mean performance of teams using deterministic and extended response threshold models of task allocation, evolved with restricted and free agent swapping	117

List of Tables

1.1	A sample of approaches to perform artificial selection, and to study the evolution of social traits	18
1.2	Conditions in which cooperative (defective) allele is under positive selection when the defective (cooperative) allele is fixated, assuming different selection pressures	21
2.1	A sample of approaches in the evolution of heterogeneous teams using individual encoding and team encoding.	49

Introduction



This thesis is concerned with the evolution of division of labor in biological and engineered multi-agent systems. Here, we set up the scene with key concepts, motivations, and methods which bind our work together. We present the main questions and challenges surrounding the evolution of division of labor and outline the method of evolutionary simulations which we used to address the ones of our interest. We emphasize the interdisciplinary context of our research by drawing parallels between natural and engineering sciences. Biologists identify behavioral mechanisms and evolutionary pressures that give rise to division of labor in social insects, whereas engineers implement functional specialization into the teams of artificial agents to increase their performance.

The ability to coordinate individual activities is fundamental to efficient operation of many engineered and biological societies. A promising strategy to control the group-level behavior is dividing the task performance among functionally specialized members. Studies of this phenomenon, named “division of labor”, span across many fields of science. In human economies, people specialize in different occupations, such as teaching, farming or trading (Smith 1776). In computer clusters, different types of computational tasks are delegated to dedicated electronic components, such as digital signal processors, graphical processing units, or field-programmable gate arrays (Topcuoglu et al. 2002). In robotics, multiple robots varying in hardware or control algorithms work together to achieve tasks typically reserved only for complex machines (Dorigo et al. in press). Last but not least, division of labor occurs at many levels of biological organization, e.g., duplicated genes often evolve to serve different functions (Rueffler et al. 2012); cells differentiate into many types (Ispolatov et al. 2012); colonies of social insects divide reproduction into fertile queens and males and sterile workers (Wilson 1971); the sterile workers further divide the labor by specializing in different activities such as nest maintenance, foraging, and brood rearing (Oster and Wilson 1978).

Division of labor is a concept unifying a diverse body of research, nevertheless different fields often ask highly specific questions. For instance, cell differentiation and human economy have little in common, although in both cases authors emphasize the importance of functional specialization. In this thesis, we narrow the scope to division of labor displayed by multiple autonomous agents working together, much like workers in an insect colony or robots in a team. With this respect, scientific effort usually follows one of the two agendas aiming to understand (i) the mechanisms and (ii) the origins of division of labor. Answering the first question involves devising models and algorithms of task allocation that explain the work distribution observed in social insects (e.g., Beshers and Fewell 2001), or induce desired global behavior in teams of artificial agents (e.g., Gerkey and Mataric 2004). The second question is usually addressed by considering the costs and benefits that specialization incurs on the society, as to understand when it evolves (e.g., Rueffler et al. 2012), or how to optimize it (e.g., Panait and Luke 2005).

The two types of questions on division are usually studied in isolation (but see Goldsby et al. 2012). However, to gain a full understanding one needs to address them simultaneously (Duarte et al. 2012). Biologists need to understand when specific mechanisms of specialization are favored by natural selection, whereas engineers need to know how to optimize the chosen task allocation algorithm. This thesis aims to rise to those challenges. To this end, by means of evolutionary simulations and formal analyses, we construct an evolutionary pathway for division of labor from the occurrence of coop-

eration, through fixed specialization, up to behavioral flexibility in task allocation. In doing so, we address several scientific and methodological questions both from biology and from engineering. In particular: How does intra-specific selection affect evolution of cooperation? How to choose a selection method in evolutionary simulations? How to design a crossover operation as to efficiently evolve fixed assignments of agents to tasks? What is the role of genetic variance in a population during team evolution? How efficient in task allocation are the existing models explaining division of labor in social insects?

In the rest of this introduction we shall review the related literature that motivated all these questions. First, we separately discuss the mechanisms and the origins of functional specialization in biological and artificial multi-agent systems. Next, we describe the method of evolutionary simulations that allow us to merge the two perspectives. We end with describing the thesis outline and contents.

Mechanisms of division of labor

Social insects perform a multitude of tasks such as feeding the brood, nest maintenance, defense, and foraging (Wilson 1971), and divide the labor among hundreds, thousands, or even millions of workers (Dornhaus et al. 2012). Similarly, engineers envision swarms of autonomous agents jointly performing complex and multiple tasks (Bonabeau et al. 1999; Eberhart et al. 2001). If so, the questions arise: What factors enable a multi-agent system to display division of labor? What mechanisms of task allocation do social insects apply, and robotic swarms should use? A common misconception is that a queen in an ant nest commands workers what to do, whereas in fact the workers are autonomous and perform tasks without any central point of control (Gordon 1996; Robinson 1992). Similarly, coordination of numerous robots operating in noisy and varying environments usually rules out centralized control algorithms. For these reasons, decentralized and self-organized division of labor in biological and engineered societies have received increased attention in the recent years.

Components of division of labor

The overall pattern of the division of labor often consists of multiple components, each stemming from a different source. Consequently, the observed level of specialization is caused by a mix of environmental, phenotypic, genetic, learning and social factors. We shall now describe them shortly, and support with empirical evidence from biology and

engineering.

To begin with, division of labor might be caused purely by environmental factors such as heterogeneous distribution of tasks in space (Johnson 2010). For example, assuming preferential performance of proximate tasks (Franks and Tofts 1994), young workers should engage in activities near their place of birth (e.g., brood rearing), and transit to tasks located outside the nest when old (e.g., foraging) (Tofts 1993). In consequence, task repertoire of workers is expected to be associated with their age, a phenomenon called “temporal polytheism”, which has been observed in many species of ants and bees (Oster and Wilson 1978; Wilson 1971). Similarly, teams of identical robots have displayed “situated” specialization (Baldassarre et al. 2003; Nouyan et al. 2009), demonstrating that homogeneous groups may divide the labor solely due to differences in the sensory-motor states caused by variation in the environment.

More often than not, specialization is owed to explicit variation in task preferences caused by phenotypic and/or genetic differences between workers. Embodied agents differing in morphology often preferentially attend to tasks best suited to their bodies. For example, around 15% of ant genera develop polymorphic workers (Gordon 1996) with small workers rearing the brood and large workers defending the nest (Robinson et al. 2009; Wilson 1980, 1985). Artificial agents may also have different bodies affecting their task repertoire, e.g., multi-robotic platform “Swarmanoid” has been composed of flying robots with the task to monitor the environment (eyebots) and ground robots with the task to transport and manipulate objects (footbots and handbots, respectively) (Dorigo et al. in press).

In the majority of biological and artificial societies agents do not display different morphologies, and yet they divide the labor due to the genetic factor. For example, in species with multiply-mated queens or multiple queens per colony, distinct lineages have been showed to differ in preferences of tasks such as guarding (Robinson and Page 1988), thermoregulation (Jones et al. 2004) and foraging (Fewell and Page Jr 2000; Ingram et al. 2005). Similarly, in engineered systems, an artificial agent may engage in selected tasks depending on the preprogrammed behavior (Luke et al. 1998). Such fixed patterns of division of labor are often discovered by optimization algorithms (Panait and Luke 2005), in particular by evolutionary simulations (Bongard 2000a; Potter and De Jong 2000). Here, we touch on this subject in Chapter 2 where we evolve fixed assignments of agents to tasks (“team compositions”).

Differences in task preferences are not always fixed, and may change due to agents’

plasticity. Labella et al. (2006) inspired by foraging behavior of ants successfully demonstrated division of labor in a team of robots by self-reinforcement in previously performed tasks (Spencer et al. 1998; Theraulaz et al. 1998). Similarly, Ravary et al. (2007) showed that ant workers tend to engage in tasks which they have experienced before. Although social insects have demonstrated remarkable learning abilities (Leadbeater and Chittka 2007), it is unclear whether the link between task preferences and plasticity goes beyond simple examples of self-reinforcement of previously performed tasks (Theraulaz et al. 1998). In contrast, in engineering, plastic specialization is often obtained through non-trivial learning algorithms often involving artificial neural networks, with applications in machine learning (Liu et al. 2000), decision making (Kohl and Miikkulainen 2009), and task allocation in multi-robotic systems (Eiben et al. 2007).

Finally, task preferences are often internally regulated by the society, showing the “social component” in division of labor. In honeybees, for example, the previously mentioned age polytheism is modulated by the demographic structure of the colony (Huang and Robinson 1996). In the absence of old bees, young bees produce more juvenile hormone and transit more quickly to foraging tasks (Beshers et al. 2001; Huang and Robinson 1992; Robinson 1987b). Similarly, the differentiation into morphologically distinct castes in ants may be a consequence of variation in nutrition that young larvae had received (Smith et al. 2008a). Other colony-level properties presumably affect specialization, too. Colony size is predicted to be positively associated with the level of division of labor (Bourke 1999; Jeanson et al. 2007), however empirical results are mixed (e.g., Dornhaus and Franks (2006); Dornhaus et al. (2009); Thomas and Elgar (2003)). Patterns of division of labor change also with the colony age (e.g., during the colony foundation more workers are involved in the nest maintenance than in the adult colony; Mailleux et al. 2003).

Task allocation algorithms

To account for various factors affecting division of labor, biologists proposed several task allocation mechanisms that social insects are argued to apply (see Beshers and Fewell 2001; Bonabeau et al. 1997b; Duarte et al. 2011; Fewell 2003; Gordon 1996, 2003; Robinson 1992 for reviews). The most prominent feature of these algorithms is that they produce patterns of division of labor which are not fixed in time by allowing for some flexibility in workers’ behaviors (Johnson 2003; Robinson 1992). There are many empirical evidence showing that external factors sensed directly through environmental cues (Bonabeau et al. 1996; Franks and Tofts 1994), or indirectly through interactions

with other workers (Adler and Gordon 1992; Gordon 1996; Pratt 2005), indeed induce workers to switch between tasks.

Similarly, artificial agents often need to dynamically change tasks in order to achieve prescribed goals. To this end, engineers have proposed many algorithms which assign agents to tasks in (nearly) optimal according to some measure, e.g., makespan or energy consumed (e.g., Berman et al. 2009; Campos et al. 2000; Dahl et al. 2009; Gerkey and Mataric 2002; Jones and Mataric 2003; Mataric et al. 2003; Schneider-Fontan and Mataric 1998). Also, multi-agent task allocation develops interesting links with other problems, such as scheduling (Gerkey and Mataric 2004), consensus building (Franceschelli et al. 2010; see Olfati-Saber et al. 2007; Ren et al. 2005 for reviews), and machine learning (Stone and Veloso 2000). Many of the algorithms used to solve such problems are inspired by the biological models of division of labor in social insects (Bonabeau et al. 1999, 2000), showing again the synergies between the two fields.

Franks and Tofts (1994) proposed the basic model called “foraging for work”, in which workers actively seek tasks and attend to the first available one. Consequently, the model describes division of labor as an emergent process owing to external factors, and with no variation among workers. It may account for some cases of age polytheism and specialization within the same caste (Tofts 1993) but fails to explain strong patterns of division of labor (Theraulaz et al. 1998). Consequently, it has been proposed to be treated as a “baseline” model denoting the level of specialization which should be expected due to spatiotemporal constraints present in the nest (Traniello and Rosengaus 1997). Similarly, in engineering, some proposed to leverage an emergent division of labor caused by spatial variation in tasks (Balch and Arkin 1998; Schneider-Fontan and Mataric 1998), or by noise in communication channel (Quinn et al. 2003).

In the “response threshold” models workers engage in a task only if the external stimuli denoting the need for its performance exceeds their internal threshold for the task (Bonabeau et al. 1996; Jeanson et al. 2007; Page Jr and Mitchell 1998; Robinson 1992). Usually, the values of thresholds are fixed during ontogeny and differ between the workers, thus reflecting the intra-colony genetic, morphological or age variation. In the “adaptive” version of the model values of the response thresholds change due to individual experience thus reflecting self-reinforcement (Theraulaz et al. 1998). Response threshold models have been successfully applied to explain colony reaction to perturbations (Waibel et al. 2006) and division of labor patterns observed in honeybees (Bertram et al. 2003), ants (Bonabeau et al. 1996), and primitively eusocial wasps (Theraulaz et al. 1998).

Due to their apparent simplicity, response threshold models have been extensively used by engineers as task allocation algorithms for groups of artificial agents (Bonabeau et al. 1999, 2000). With this respect there are, however, two issues. First, there is a question of how to find appropriate values of the thresholds. Biologists often randomize the thresholds (e.g., Jeanson et al. (2007)) which, however, may impair the overall performance of the society. To overcome this problem, engineers optimize the values of thresholds with learning algorithms (Labella et al. 2006), or by means of evolutionary simulations (Campos et al. 2000). We touch on this subject in Chapter 2 where we evolve fixed assignments of agents to tasks, and more in Chapter 3 where we evolve the values of response thresholds in a foraging scenario.

The second problem is that response threshold models originally assumed that tasks stimuli are commonly available to every worker (Bonabeau et al. 1996), which is possible only when the needs of the society may be efficiently communicated and aggregated by the members. This is the case of multi-agent systems with adequate communication capabilities, like teams of robots with a central unit overseeing their behavior (Berman et al. 2009; Krieger et al. 2000). Also, response thresholds have been used to solve scheduling (Bonabeau et al. 1997a; Campos et al. 2000) and load balancing (Cao 2004; Freeman et al. 2006) problems in computational clusters where communication between computer processes is relatively easy. Alternatively, global needs may sometimes be inferred from locally accessible information (Agassounon and Martinoli 2002; Jones and Matarić 2003; Matarić et al. 2003, but see Parker 1993).

To address the issue of global stimuli, biologists developed “networking” models of division of labor, which focus on information transfer through the colony by intra-specific cues and signals (Fewell 2003; Gordon 1996; Gordon et al. 1992; Pacala et al. 1996; Solé et al. 1993). In these models, workers choose tasks in response to inter-workers interactions. For example, individuals from inside the nest have been observed switching to foraging if recruited by patrollers who successively located the food source (Greene and Gordon 2007b; Richardson et al. 2010), or if successful foragers come back at a precise rate (Gordon et al. 2008; Greene and Gordon 2007a; Schafer et al. 2006). Of note, “networking” models often emphasize the similarity between a nervous system and an insect society as if the colony were a big neuronal network with single workers mimicking neurons (Hirsh and Gordon 2001; Seeley et al. 2012).

Message passing between neighboring agents is also an often used technique in task allocation in robotic systems (Dahl et al. 2009; Parker and Zhang 2009). The most well-known family of such methods are market-based algorithms (Gerkey and Matarić

Introduction

2002; Lagoudakis et al. 2005; Tovey et al. 2005), where agents compare their desire to perform a task (“bid”) and the most eager agent actually performs it (“wins the auction”). The market-based algorithms are similar to the ones using response thresholds, in the sense that with the former approach agents compare their will to perform the task with each other, whereas in latter approach they compare it with their internal thresholds (see also Kalra and Martinoli 2006).

Origins of division of labor

Mechanistic explanations of division of labor answer “how” specialization may be achieved, but say little about “why” it occurs, evolves, or should be implemented. Although specialization may be an emergent process (e.g., Franks and Tofts 1994), more often than not it is the effect of evolutionary pressures, or engineering decisions. Dividing activities among specialized workers is presumably the main source of high ecological success of social insects (Hölldobler and Wilson 1990; Oster and Wilson 1978; Robinson 1992). Similarly, engineers create teams of specialized artificial agents hoping it will increase the system’s performance (in comparison to a group of non-specialists Panait and Luke 2005). If so, questions arise: What benefits are associated with division of labor? Are there any costs? And more precisely, what evolutionary pressures and engineering guidelines favor or prevent functional specialization?

Benefits and costs of division of labor

The effects of division of labor on societies have been first considered by Smith (1776), the pioneer of economical science. He speculated that division of labor benefits human industries in three ways: (i) increases individual efficiency, (ii) avoids the costs of task switching, and (iii) facilitates the invention of machines. These three arguments made their way into biology and engineering, however each to a different extent.

The first argument raised by Smith (1776) states that specialized individuals become masters of their trades and increase the quality and the quantity of their work. This is often true in engineered systems. For example, in heterogeneous computational clusters each type of electronic devices is dedicated to a different computational task, in which it greatly outperforms general purpose computers (Bader et al. 2009). In the RoboCup challenge, robots increased their individual efficiency by specializing in some aspect of the soccer game, like goalkeeping and shooting. Consequently, teams that divided the labor beat the ones that did not (Luke et al. 1998). In contrast to engineered systems, the

evidence that division of labor increases individual efficiency in insects colonies is mixed. In ants with morphological castes, smaller workers may be indeed more efficient in brood care, whereas larger workers may be better at nest defense and foraging (Gordon 1996). However, without differences between workers in morphology or size, individual experience does not affect the efficiency in task performance (Dornhaus 2008), despite it may change the task preferences (Ravary et al. 2007).

The second argument raised by Smith (1776) states that with each task being handled by a distinct group of individuals, no one needs to switch between the tasks thus avoiding the associated costs, such as cognitive overhead and the time lost in passing from one activity to another (Sendova-Franks and Franks 1995). The avoidance of task switching costs has been argued to be an important evolutionary pressure that give rise to division of labor in social insects (Dornhaus 2008). This hypothesis has been recently supported by means of computer simulations which demonstrated that higher costs of task switching lead to the evolution of stronger division of labor in colonies of digital organisms (Goldsby et al. 2012). Additionally, in engineering, switching between tasks incurs costs related to actually building agents capable of multi-tasking, which might prove challenging. Decomposing a single complex robot into a team of specialized simpler units is an alternative and promising approach in designing robotic systems. For example, the previously mentioned Swarmanoid system was the first to successfully demonstrate a team of relatively simple robots which divided the labor and collectively performed human-like tasks, such as reaching for a book on a shelf. By designing the system as a set of functionally specialized robots, engineers avoided the difficulties (costs) of building complex humanoid robot (Dorigo et al. in press).

The final third argument raised by Smith (1776) states that division of labor facilitated the invention of machines because specialization led to partitioning of complex activities into many small tasks, which then became easier to automate. This argument is specific to human industries and to our knowledge has not been adopted by biologists, nor by engineers. However, radical forms of workers' polymorphism might be interpreted as the equivalence of inventing machines. For example, in honeypots ants, some workers are overfed and their abdomens swell which turns them into living containers of food, from which other ants nourish (Ratnieks and Anderson 1999). It might be speculated that previous weaker forms of division of labor opened the doors for such strong specialization.

Functional specialization entails many other benefits that go beyond the original list proposed by Smith (1776). For example, the evolution of age polytheism may be

Introduction

explained by variation in mortality risks associated with tasks (O'Donnell and Jeanne 1995; Wakano et al. 1998). Foraging is more death threatening than brood rearing due to higher activity of predators outside than inside the nest. Since production of workers requires high energetic expenses (Hou et al. 2010), thus to maximize the utility derived from each worker they should engage in task activities until the end of their natural lifespan. Consequently, young workers should perform tasks with low mortality risk (inside the nest), whereas old workers may transit to tasks with high mortality risks (outside the nest). Such reasoning provides an evolutionary explanation of the previously mentioned temporal polytheism displayed by social insects. A yet another example is "task partitioning", when a single complex task is divided into two or more that are handled by specialized workers (Ratnieks and Anderson 1999). In such a case, the benefit of division of labor stems from eliminating the preexisting constraint on effective performance of the original complex task. In engineering, perhaps the most vivid example of task partitioning is the invention of the assembly line by Henry Ford.

Although division of labor is a widespread strategy to control the group-level behavior, it has not evolved and has not been implemented in all instances of biological and engineered societies. To identify when division of labor is under negative selection, or should be avoided, one needs to understand the costs it incurs. Already Smith (1776) noticed that division of labor might impair the well-being of workers due to mundane repetitions of the same task over and over again. Most importantly, high levels of specialization and task partitioning often entails a loss in system's robustness against external perturbations (Ratnieks and Anderson 1999; Rueffler et al. 2012; Waibel et al. 2006), a point valid for both biological and engineered systems. In social insects, for example, fixed division of labor decreases the overall performance when needs of a colony change during ontogeny (Waibel et al. 2006). It has been argued that social insects overcome this issue by producing an excess of workers, which remain inactive and engage in task performance only when mobilized (Robinson 1992). Similar arguments are raised with respect to robustness against failure in robotic swarms (Şahin 2005).

In technical sciences, there are two other costs related to engineering functional specialization in societies of artificial agents. First, building platforms composed of specialized subunits may impose additional maintenance costs related to, e.g., hardware integration. Second, many argued that it is usually more challenging to optimize multiple distinct controllers than a single one applied by all agents in a team (Panait and Luke 2005). However, in contrast, functionally specialized agents often require simpler control mechanisms than generalists due to reduced behavioral repertoire (Bongard 2000b). This tradeoff between intra- and inter-agent complexity remains a big open

question in the field of team optimization (see Chapter 2 for further information).

Beyond cooperative societies

So far, we took a group-level perspective and considered benefits received and costs paid by the entire society due to functional specialization of its members. But to display a stable pattern of division of labor, these members must cooperate with each other. For example, human industries are constrained by job contracts, and artificial agents are often designed to work together by default. In contrast, in biology, evolution of cooperation has been considered puzzling (Colman 2006) and has attracted extensive scientific interest (see Griffin et al. 2004; Lehmann and Keller 2006; Robinson et al. 2005; Sachs et al. 2004; West et al. 2006 for reviews). This is because, although cooperation is often beneficial, it is also sensitive to cheating. To illustrate this, consider a cooperative act of one individual towards the other that infers a fitness cost to the donor, and a fitness benefit to the recipient. The recipient of the cooperative act who does not cooperate himself (defects) increases his fitness at the expense of others. For example, in a pack of predators that share the prey, a defector fakes the participation and benefits easy access to the food while not risking the dangers of hunting. Defectors gain a survival advantage over cooperators, they spread in the population and cooperation collapses. Thus, even if the animals in the pack have had specialized in different parts of the hunt, this division of labor would collapse along with cooperation.

Various mechanisms may oppose the spread of defectors and facilitate cooperation (see Lehmann and Keller 2006; Nowak 2006b for overviews). For example, punishment of defection (Fowler 2005; Hamilton 1970; Ratnieks et al. 2006) renders it to be an unrewarding behavior, whereas positive assortment between cooperators limits the number of defective interactions. The positive assortment might be caused by, e.g., relatedness between interacting individuals (Foster et al. 2006; Hamilton 1964; Queller 1992), or by reciprocity towards cooperators (Queller 1985; Rand et al. 2009; Trivers 1971). Indeed, many cases of division of labor have been reported in genetically identical multi-component systems like multi-cellular organisms in which there are no reproductive conflicts (Rueffler et al. 2012). Similarly, insect species known to divide the labor (ants, bees, and termites) are highly social and live in family-based colonies, supporting the idea that the relatedness indeed plays a crucial role in the evolution of cooperation (Hamilton 1964). Consequently, only the societies of highly related members evolve functional specialization, as only they can withstand the invasion of selfish individuals. We touch more on this subject in Chapter 1 where we show how mechanisms of intra-

specific selection regulate the evolution of cooperation.

Evolutionary synthesis of mechanisms and origins

We have discussed the mechanisms and the effects of functional specialization in multi-agent systems rather independently from each other. To fully understand division of labor one must inevitably merge the two perspectives (Duarte et al. 2011). Biologists need to understand the evolutionary pathways leading to specific mechanisms enabling division of labor observed in nature. Engineers need to know how to optimize task allocation algorithms, which they have crafted. These two are very challenging problems but both may be addressed (to certain extent) by means of simulations of evolution performed in a computer. These computational methods are at the heart of this thesis. Here, we provide a short introduction to evolutionary simulations, for a more detailed description and applications in biology and engineering see Sections 1.3 and 2.3, respectively.

Originally, evolutionary simulations were proposed to automatically find solutions to computational problems, much like evolution discovers new adaptive traits (Fogel 1994; Fogel et al. 1966). Evolutionary simulations usually operate on a finite population of genotypes, where each genotype encodes a solution to a given problem. For example, in evolutionary robotics a genotype is a sequence of parameters, which define the control system of a robot or its morphology (Floreano and Keller 2010). These artificial genotypes are subjects to differential selection, mutation, and/or recombination. Genotypes encoding better solutions are selected more often than genotypes encoding worse solutions. Consequently, fitter genotypes spread in the population and artificial evolution occurs (Adami 2006; Smith 1992).

Evolutionary simulations have been popularized by many practitioners (e.g., De Jong 2007; Goldberg 1989; Holland 1992; Koza 1992; Mitchell 1996) and are mainly applied to problem solving. In particular, they have been successfully used to optimize group-level behavior in teams of artificial agents (Panait and Luke 2005). Nevertheless, evolution of teams remains a challenging problem with many open questions (see Chapter 2 for more details). In parallel, evolutionary simulations become increasingly popular in biological studies (Foster 2001; Smith 1992; Wagner and Altenberg 1996), mainly because they give precise control over the experimental conditions (Floreano and Keller 2010) and allow the study of evolution at unprecedented level of detail (Adami 2006). In particular, they have been used to address challenging biological questions,

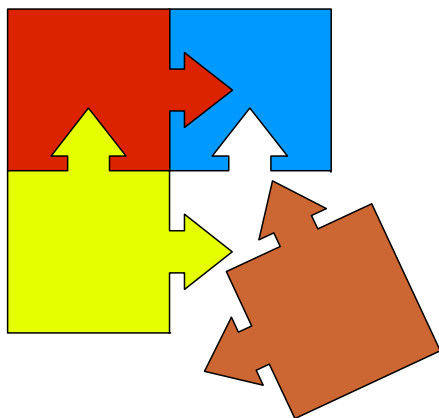
including those related to division of labor (Duarte et al. 2012; Tarapore et al. 2010; Waibel et al. 2006; see Chapters 1 and 3 for more details).

Thesis outline

The rest of this thesis is composed of the three chapters containing the main results and the conclusions summarizing our main contributions and pointing to future direction of research. All three main chapters contain material submitted to or published in scientific journals (see Sections 1.1, 2.1, and 3.1 for more details). Consequently, all chapters follow the same, article-like structure composed of five main parts: (1) Introduction presenting state of the art and main motivations of a chapter; (2) Methods explaining experimental settings and modeling details; (3) Results with novel contributions; (4) Discussion putting the results in a broader context; and (5) Supplementary materials with formal proofs and additional analyses.

In Chapter 1, we use evolutionary simulations of artificial agents with behavior reduced to a binary choice between cooperation and defection (no cooperation). We show theoretically and experimentally that the mechanisms of intra-specific selection regulate the evolution of cooperation. Knowing the conditions that enable cooperation and open the doors for division of labor, in the two following chapters we focus on cooperative societies. In Chapter 2, we explore the ways to optimize genetically fixed specialization by means of evolutionary simulations. We show how to control the genetic variance in the population, and consequently make the evolutionary search find globally optimal assignments of agents to tasks (“team compositions”). Finally, in Chapter 3, we study functional specialization at the behavioral level in simulated ant colonies. We quantify and compare the efficiency and flexibility in task allocation between several commonly used biological models of division of labor. We show theoretically and experimentally that these models fail to explain precise reallocation of workforce in response to changes in the environment.

1 Selection methods regulate evolution of cooperation in artificial evolution



To begin with, members of a society must cooperate with each other in order to divide the labor. Biologists have identified many factors which facilitate the evolution of cooperation such as, for example, a positive relatedness level between the society members. Here, we contribute to the understanding of the origins of cooperation by demonstrating theoretically and experimentally that its evolution is regulated by mechanisms of intra-specific selection. We, thus, show a novel pathway to cooperation, and to the potentially following division of labor which we will study in the next two chapters. From a methodological standpoint, we expose that deciding on a particular “selection method” that chooses which individuals survive during evolutionary simulations constitutes a crucial step in the modeling process as it has important implication on the evolutionary outcome of the investigated traits.

1.1 Disclosure

Contents of this chapter are taken from an article of the same title, which have been recently submitted to a scientific journal. The article is authored by Paweł Lichocki, Dario Floreano and Laurent Keller. Paweł Lichocki developed formal analyses, implemented and executed simulations, and wrote the manuscript. All authors designed the project, discussed the results and revised the manuscript. Steffen Wischmann and Barbara Piasecka contributed useful comments on the manuscript. The work was supported by the Swiss National Science Foundation and an ERC advanced grant.

1.2 Abstract

A key, yet often neglected, component of artificial evolution is the “selection method” which assigns fitness (number of offspring) to individuals based on their performance scores (efficiency in performing tasks). Here, we formally and experimentally study the evolution of cooperation under the five most common selection methods (proportionate, rank, truncation-proportionate, truncation-uniform, and tournament). We consider related individuals engaging in a prisoner’s dilemma game where individuals can either cooperate or defect with cooperation’s costs and benefits affecting their performance scores. We show that cooperation was positively associated with the relatedness between interacting individuals under all methods. However, the change in the performance benefit of cooperation affected the populations’ average level of cooperation only under the proportionate selection methods. Truncation and tournament methods introduced negative frequency-dependence and led to the evolution of polymorphic populations. This study reveals that the mode of selection plays an important role in regulating the evolution of cooperation.

1.3 Introduction

Researchers address evolutionary questions with various methods ranging from mathematical models to wet-lab and field experiments. These approaches are highly successful, but have limitations. For example, mathematical models make simplifying assumptions about complex ecological interactions in order to be tractable (Schoener 2011). Long-term evolutionary experiments with organisms having generation time higher than bacteria are practically impossible (Adami 2006). Artificial evolution performed with computer simulations have been advocated as an alternative and promising approach to bypass

such limitations (Adami 2006; DeAngelis and Mooij 2005; Floreano and Keller 2010; Foster 2001; Mitri et al. 2013; Smith 1992; Wagner and Altenberg 1996). An evolutionary simulation operates on a finite population of individuals (Fogel 1994), each having a genome encoding its morphology and/or behavior. The “selection method” determines on the basis of individual performance which individuals will contribute offspring, after selection and/or recombination, to the next generation.

Several selection methods are being used in studies of artificial evolution with computer simulations. The proportionate selection method (PSM) chooses the individuals contributing to the next generation proportionally to their performance scores (Goldberg 1989). The rank selection method (RSM) chooses a parent proportionally to the ranks (positions in a sequence of individuals sorted ascending by the performance scores) (Mitchell 1996). With both PSM and RSM any individual has a chance to contribute to the next generation. In contrast, with the “truncation” methods of selection only a certain fraction of the population (i.e., the best performing individuals) contributes offspring to the next generation. The truncation-proportionate selection method (TPSM) chooses a parent proportionally to performance scores, whereas the truncation-uniform selection method (TUSM) chooses a parent uniformly at random (Back 1994; Crow and Kimura 1979; Crow et al. 1970; Milkman 1978; Schlierkamp-Voosen 1993). Finally, the tournament selection method (TSM) forms “tournaments” by sampling individuals with replacement uniformly at random from the entire population. The genotypes of the individuals with the highest performance scores in each tournament are copied to the descending generation (Blickle and Thiele 1995; Goldberg and Deb 1991).

While all five selection methods are frequently used to simulate differential selection (Table 1.1.A), the choice between them is rarely justified. Moreover, little attempt has been made to quantify the effects of selection methods on the dynamics of the evolutionary simulation (but see Ficici et al. 2005; Hauert and Doebeli 2004). This is a major issue because each selection method defines a different mapping between performance scores and fitness (Appendix A), thus having important implications on the course of evolution (Gillespie 1975, 1977; Lehmann and Balloux 2007; Rice 2008). To address this problem, we investigate theoretically and in simulations how the five selection methods regulate the evolution of cooperation. We focus on cooperation because evolutionary simulations are especially popular in this domain (Table 1.1.B), and it is an important biological phenomenon that has attracted extensive scientific interest (see Griffin et al. 2004; Lehmann and Keller 2006; Robinson et al. 2005; Sachs et al. 2004; West et al. 2006 for reviews). We consider a population of related individuals, each having a genotype that consists of a haploid allele encoding for cooperation or defection.

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

Table 1.1: A sample of approaches to perform artificial selection (a), and to study the evolution of social traits by means of evolutionary simulations (b).

(A) Selection method	References
Proportionate selection method (PSM)	Meuleau and Lattaud (1996), Agah and Bekey (1997), Bowles and Gintis (2004), Hauert and Doebeli (2004), Leimar and Hammerstein (2001), Ward et al. (2001), Marocco et al. (2003), Doherty and O’Riordan (2007), Waibel et al. (2009), Waibel et al. (2011), Mitri et al. (2011), Duarte et al. (2012), Goldsby et al. (2012), Niv et al. (2002)
Rank selection method (RSM)	Quinn et al. (2003), Wischmann et al. (2012)
Truncation-proportionate selection method (TPSM)	Tarapore et al. (2010), Lichocki et al. (2012), Tuci et al. (2002)
Truncation-uniform selection method (TUSM)	Nolfi and Floreano (1998), Cangelosi and Parisi (1998), Cangelosi (2001), Baldassarre et al. (2003), Waibel et al. (2006), Floreano et al. (2007), Mitri et al. (2009), Agogino et al. (2000), Kashtan et al. (2007)
Tournament selection method (TSM)	Huberman and Glance (1993), Riolo et al. (2001), Hauert and Doebeli (2004)
(B) Domain	References
Cooperation	Meuleau and Lattaud (1996), Agah and Bekey (1997), Ward et al. (2001), Leimar and Hammerstein (2001), Bowles and Gintis (2004), Hauert and Doebeli (2004), Doherty and O’Riordan (2007), Waibel et al. (2009), Waibel et al. (2011), Mitri et al. (2011), Axelrod (1987), Nolfi and Floreano (1998), Fogel (1993), Huberman and Glance (1993), Riolo et al. (2001), Quinn et al. (2003), Baldassarre et al. (2003)
Communication	Marocco et al. (2003), Quinn (2001), Cangelosi and Parisi (1998), Cangelosi (2001), Floreano et al. (2007), Mitri et al. (2009), Wischmann et al. (2012)
Division of labor	Duarte et al. (2012), Goldsby et al. (2012), Waibel et al. (2006), Tarapore et al. (2010), Lichocki et al. (2012)

The individuals engage in a social game of prisoner’s dilemma (Eshel and Cavalli-Sforza 1982; Smith and Price 1973) where a cooperator pays a cost and its partner receives a benefit. Because individuals only interact once, a cooperative trait is effectively altruistic as it decreases the lifetime performance and fitness of the individual performing it (Hamilton 1964; Lehmann and Keller 2006). In mathematical models, the cost and benefit of cooperation directly affect the fitness of the individuals (Nowak 2006a; Rice 2004). We extend this approach by considering that the cost and benefit of cooperation

affect performance scores, which are translated into fitness by one of the five selection methods (PSM, RSM, TPSM, TUSM, and TSM). For each selection method, we formally identify the conditions in which cooperation evolves and we experimentally quantify its level.

1.4 Methods

1.4.1 Selection methods

All selection methods, except for TSM, constructed the descending generation in a population of n individuals by independently sampling with replacement n individuals from the parenting generation. With PSM the probability of sampling the individual i was equal $f_i / \sum_{j=1}^n f_j$, where f_i was the performance score of the individual i . With RSM the probability of sampling the individual i was equal $r_i / \sum_{j=1}^n r_j$, where r_i was the rank of the individual i , i.e., its position in the sequence of all individuals sorted ascending by performance scores (individuals with the same performance score had the same rank). With TPSM and TUSM only the t fraction of individuals with the highest performance scores in the population were considered viable. Let S_t denote the set of viable individuals' indices. With TPSM the probability of sampling the individual i was equal $f_i / \sum_{j \in S_t} f_j$ if $i \in S_t$, and 0 otherwise. With TUSM the probability of sampling the individual i was equal $1/|S_t|$ if $i \in S_t$, and 0 otherwise. Finally, TSM with the tournament size s constructs the descending generation in a population of n individuals by selecting the winners of n independent tournaments. In each tournament, s individuals were sampled with replacement uniformly at random from all n individuals of the parenting generation. The tournament's winner was the individual with the highest performance score among the s individuals, and ties were resolved uniformly at random.

1.4.2 Model of cooperation

Each individual interacted with probability r once in its lifetime with itself, and with another individual chosen uniformly at random, otherwise (Eshel and Cavalli-Sforza 1982). We kept the population size constant, thus, r reflected the average genetic relatedness between the individuals (Zhang and Hui 2011). Depending on the outcome of the interaction, each individual received a performance score equal to the payoff of the normalized linear prisoner's dilemma game (Smith and Price 1973). A cooperator received performance score B if it interacted with a cooperator, and 0 if it interacted with a defector. A defector received performance score $B + C$ if it interacted with a cooperator,

and C if it interacted with a defector.

1.4.3 Evolutionary experiments

We evolved a population of 1000 individuals in 11 treatments. In each treatment we used a different selection method: PSM, TPSM with truncation threshold $t = 0.8, 0.5, 0.2$, TUSM with truncation threshold $t = 0.8, 0.5, 0.2$, TSM with tournament size $s = 2, 3, 5$, and RSM. For each treatment we investigated 40×51 conditions, with the performance cost of cooperation C fixed to 1, the performance benefits of cooperation B ranging between 1.1 and 5 with a step of 0.1, and the relatedness level r ranging between 0 and 1 with a step of 0.02. For each treatment and each condition we replicated the numerical experiment 30 times. Each evolutionary experiment lasted for 1000 generations. Population size, and the values of B and r were kept constant across generations.

1.4.4 Genetic architecture, selection and reproduction

Each individual had a genotype consisting of one binary allele denoting the lack (0) or the possession (1) of the cooperative trait. At the first generation of each evolutionary run all 1000 individuals had the allele set to 0. At each generation, the parenting individuals were entirely replaced by their offspring. The individuals received performance scores of the normalized linear prisoner's dilemma game with fixed performance cost of cooperation C equal 1. Consequently, a cooperator (defector) received performance score B ($B + 1$) if it interacted with a cooperator, and 0 (1) if it interacted with a defector. To construct the descending generation, 1000 individuals were chosen by a selection method that depended on the treatment. Finally, the genotypes of the selected individuals were copied to the descending generation, and mutated (with probability 0.001 the value of an allele was flipped).

1.4.5 Statistical analysis

For each replicate, treatment and condition, we measured the evolved cooperation level as the proportion of cooperators in a population averaged across generations 900 – 1000. To compare the selection methods, we considered mean cooperation level over all conditions (Fig. 1.1). To investigate the effect of the relatedness level r , we quantified the average cooperation level over all conditions with the same value of r (Fig. 1.2, red line). To investigate the effect of the performance benefit of cooperation B , we quantified the average cooperation level over all conditions with the same value of B (Fig.

1.2, blue line). In addition, we compared the results of formal analyses and numerical simulations (Figures 1.3 and 1.4). Finally, for each replicate, treatment and condition, we measured the stability of the evolved population as the mean absolute difference between proportions of cooperators in a population in consecutive generations, averaged across generations 900 – 1000 (Fig. 1.5). Statistical significance between all treatments was determined with Kruskal-Wallis test (nonparametric one-way analysis of variance) and between a pair of treatments with Wilcoxon test (rank sum test for equal medians).

1.5 Results

1.5.1 Formal analyses

We formally investigated the spread of a cooperative allele in a population of individuals related at level r on average. Individuals interact in pairs, and a cooperator pays a cost $C > 0$ which is subtracted from its performance score, whereas the partner of a cooperator receives a benefit $B > C$ which is added to its performance score. The performance scores of the individuals are translated into their fitness values by one of the five selection methods: PSM, RSM, TPSM, TUSM, and TSM. Two of them (PSM and RSM) do not have any free parameters. In contrast, TPSM and TUSM are characterized by the truncation threshold t , which determines the fraction of individuals (i.e., those with the highest performance scores) which are viable. Finally, TSM is characterized by the tournament size s , which indicates the number of individuals that compete between each other in randomly formed groups. Using each of the five selection methods we identified the

Table 1.2: Conditions in which (A) the cooperative allele is under positive selection assuming the defective allele is fixated, and (B) *vice versa*. r denotes the relatedness level between individuals, B is the performance benefit of cooperation, and C is the performance cost of cooperation. The performance scores of the individuals were transformed into fitness by one of the five selection methods: proportionate (PSM), rank (RSM), truncation-proportionate with truncation threshold t (TPSM t), truncation-uniform with truncation threshold t (TUSM t), and tournament with tournament size s (TSM s).

Selection	(A) Cooperation invades defection when	(B) Defection invades cooperation when
PSM	$r > C/B$	$r < C/B$
RSM	$r > 1/2$	$r < 1/2$
TPSM t	$r > tC/B$	$r < 1 - tB/(B + C)$
TUSM t	$r > t$	$r < 1 - t$
TSM s	$r > 1/s$	$r < 1 - 1/s$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

conditions for the evolution of cooperation (Table 1.2.A) and defection (Table 1.2.B). In general, these conditions depend on the frequency of cooperators in a population (*Supplementary materials*, Sections 1.7.1-1.7.5). Thus, we focused on the conditions when the invading allele is under positive selection, assuming that the opposite allele is fixated.

With PSM, cooperation is under positive selection in a population of defectors when $r > C/B$, whereas defection is under positive selection in a population of cooperators when $r < C/B$. With TPSM these conditions are relaxed, and cooperation is under positive selection in a population of defectors when $r > tC/B$, whereas defection is under positive selection in a population of cooperators when $r < 1 - tB/(B + C)$. In contrast to PSM and TPSM, with RSM, TUSM, and TSM the conditions for invasion of cooperation and of defection are independent of C and B . Cooperation is under positive selection in a population of defectors when $r > 1/2$ for RSM, $r > t$ for TUSM, and $r > 1/s$ for TSM. Similarly, the defection is under positive selection in a population of cooperators when $r < 1/2$ for RSM, $r < 1 - t$ for TUSM, and $r < 1 - 1/s$ for TSM. All results are jointly presented in Table 1.2. For formal derivations and extended analyses see (*Supplementary materials*, Sections 1.7.1-1.7.5).

1.5.2 Simulations

To verify and extend our formal analyses, we experimentally quantified by means of computer simulations the average cooperation level in 30 populations under each of the five selection methods. Overall, there were eleven treatments because we systematically investigated different values of the truncation threshold t and of the tournament size s (PSM, RSM, TPSM $t = 0.8, 0.5, 0.2$, TUSM $t = 0.8, 0.5, 0.2$, TSM $s = 2, 3, 5$). For simplicity, we fixed the performance cost of cooperation $C = 1$, and investigated the combined effects of the relatedness level r and the performance benefit of cooperation B on the cooperation level. There were significant differences in the level of cooperation averaged across all conditions between all eleven treatments (Fig. 1.1, Kruskal-Wallis test, $df = 10$, $p < 0.001$) and between each pair of treatments (55 pairwise Wilcoxon tests, $df = 29$, all $p < 0.001$).

The performance benefit of cooperation B had different effects on the level of cooperation depending on the selection method used. The cooperation level increased with B in the four treatments with PSM and TPSM $t = 0.8, 0.5, 0.2$. By contrast, in the five other treatments the value of B had no effect on the level of cooperation (Fig. 1.2, blue line).

In all 11 treatments, the level of cooperation increased with relatedness r (Fig. 1.2,

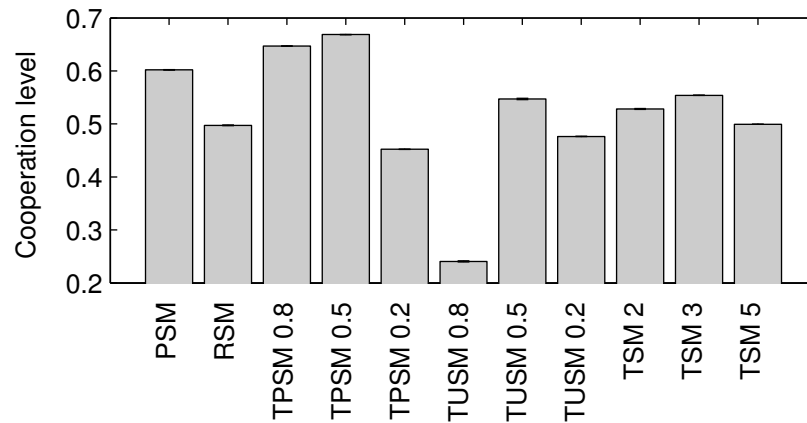


Figure 1.1: Mean \pm s.d. cooperation level over all conditions (30 replicates). There were 11 treatments, and in each a different selection method was used: proportionate (PSM), truncation-proportionate with threshold $t = 0.8$ (TPSM 0.8), $t = 0.5$ (TPSM 0.5), and $t = 0.2$ (TPSM 0.2), truncation uniform with threshold $t = 0.8$ (TUSM 0.8), $t = 0.5$ (TUSM 0.5), and $t = 0.2$ (TUSM 0.2), and tournament with size $s = 2$ (TSM 2), $s = 3$ (TSM 3), and $s = 5$ (TSM 5).

red line). However, there were differences among treatments about the nature of the transition from defection to cooperation. There was a thresholding effect in the four treatments with RSM, TUSM $t = 0.8, 0.5$, and TSM $s = 2$ as the evolved populations contained either defectors (for low values of r) or cooperators (for high values of r ; Fig. 1.2.BFGI, red line). By contrast, in the seven other treatments the transition from defection to cooperation with the increase of r was gradual. This was because the cooperation level depended on combined effects of r and B with PSM and TPSM $t = 0.8, 0.5, 0.2$ (Fig. 1.3.ACDE), and because polymorphic populations evolved with TPSM $t = 0.8, 0.5, 0.2$, TUSM $t = 0.2$, and TSM $s = 3, 5$ (Fig. 1.3.CDEHJK).

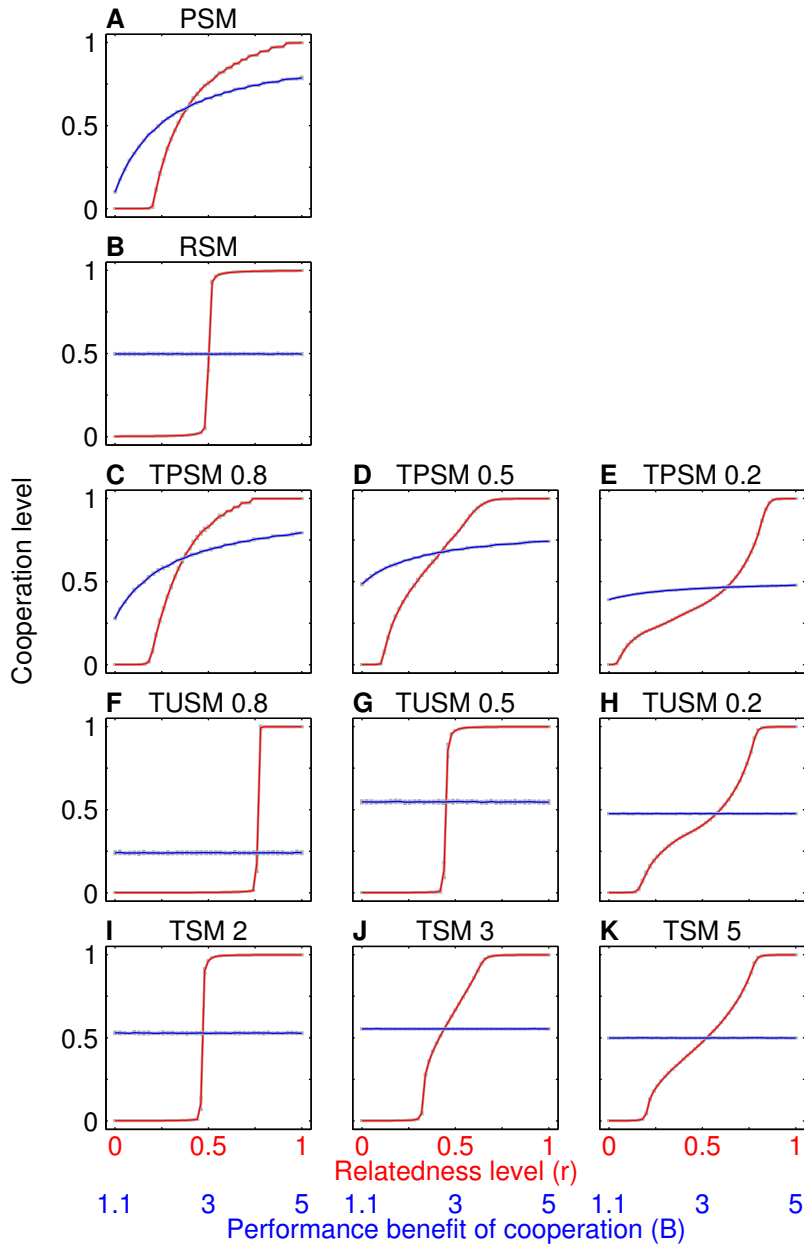


Figure 1.2: Mean \pm s.d. (in grey) cooperation level over: (blue) all conditions with the same performance benefit of cooperation, (red): all conditions with the same relatedness level (30 replicates). There were 11 treatments, and in each a different selection method was used: proportionate (A), rank (B), truncation-proportionate with threshold $t = 0.8$ (C), $t = 0.5$ (D), and $t = 0.2$ (E), truncation uniform with threshold $t = 0.8$ (F), $t = 0.5$ (G), and $t = 0.2$ (H), and tournament with size $s = 2$ (I), $s = 3$ (J), and $s = 5$ (K).

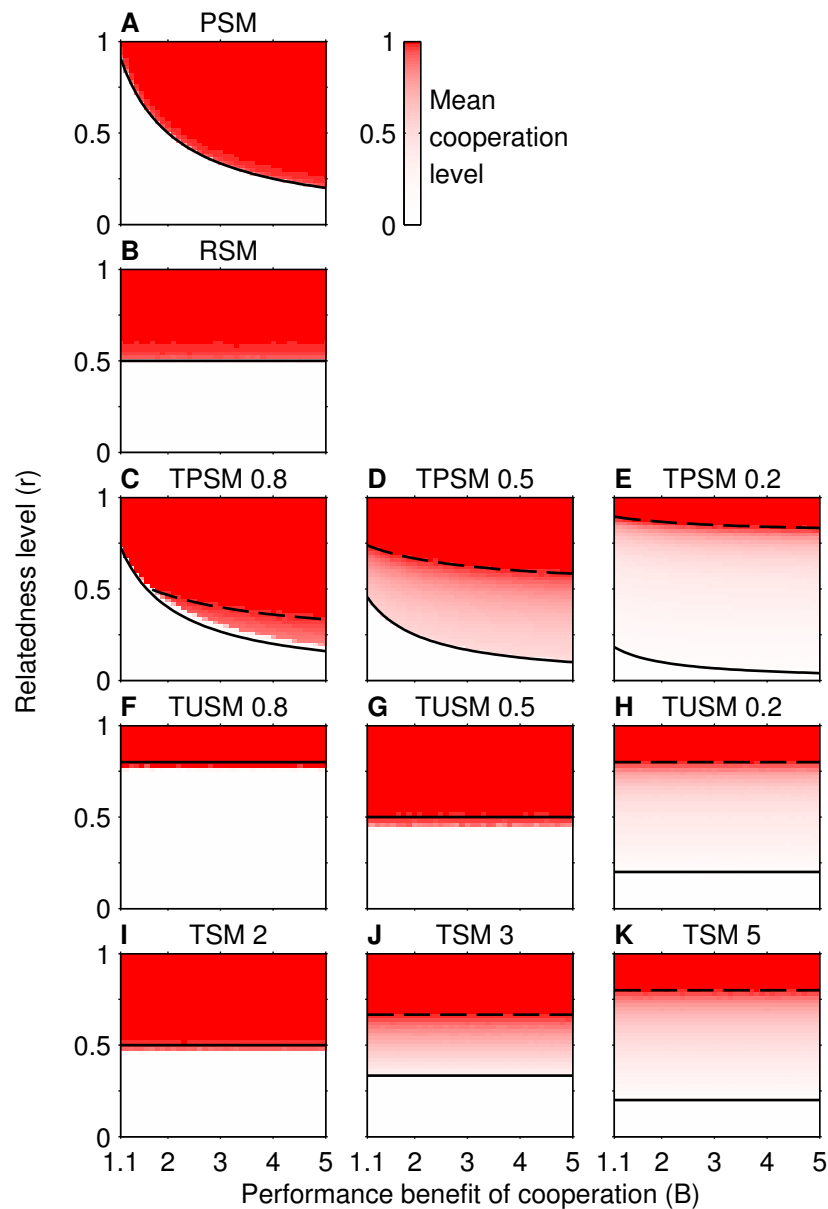


Figure 1.3: Mean cooperation level over 30 replicates for every condition. See caption of Fig. 1.2 for the explanation of the treatments. Cooperation was always under positive selection (irrespective of the proportion of cooperators and defectors in the population) in conditions above dashed line. Defection was always under positive selection (irrespective of the proportion of cooperators and defectors in the population) in conditions below solid line. Either cooperation or defection was under positive selection depending on the proportion of cooperators in a population in conditions above the solid line and below the dashed line (compare with Table 1.2).

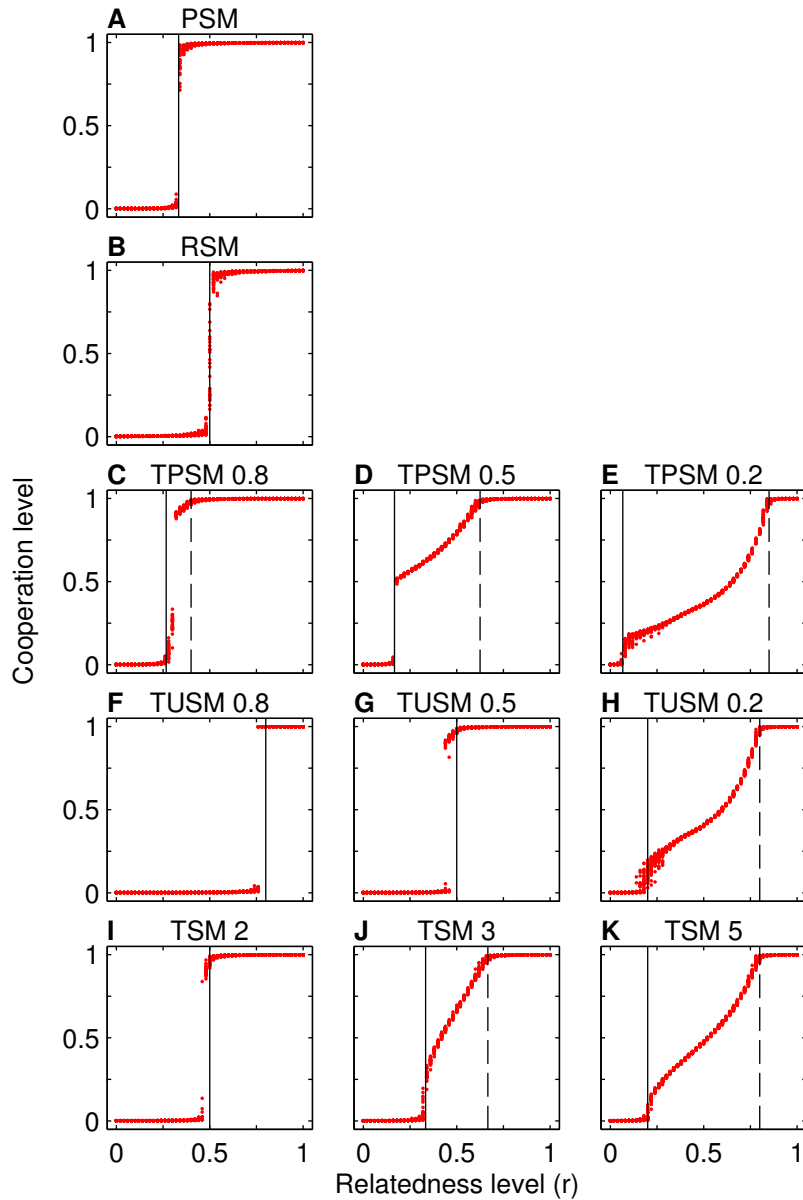


Figure 1.4: Cooperation level in each of the 30 replicates for conditions with $B = 3$. See caption of Fig. 1.2 for the explanation of the treatments. Cooperation was always under positive selection (irrespective of the proportion of cooperators and defectors in the population) in conditions to the right of the dashed line. Defection was always under positive selection (irrespective of the proportion of cooperators and defectors in the population) in conditions to the left of the solid line. Either cooperation or defection was under positive selection depending on the proportion of cooperators in a population in conditions to the right of the solid line and to the left of the dashed line (compare with Table 1.2).

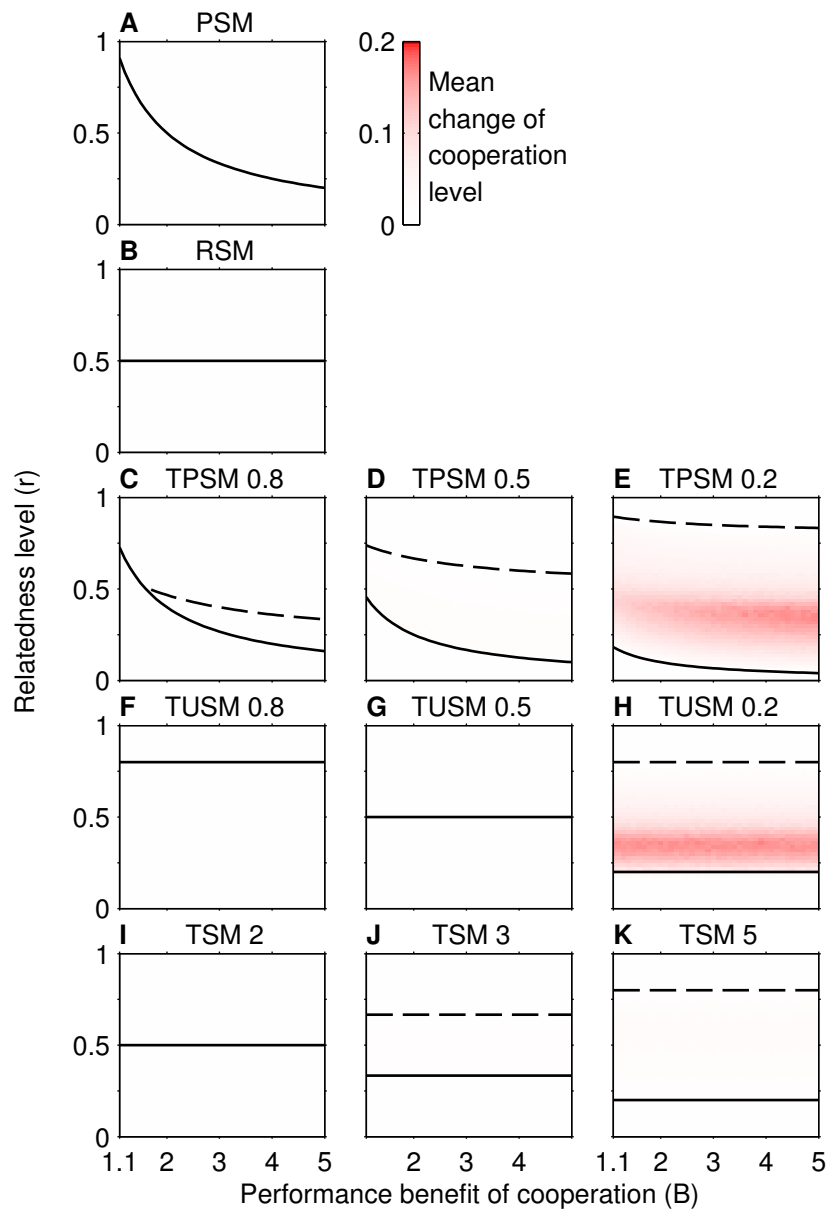


Figure 1.5: Mean absolute rate of change of the proportion of cooperators in a population between the current and the next generation (averaged across generations 900-1000 and over 30 replicates). See caption of Fig. 1.2 for the explanation of the treatments.

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

For all treatments, the simulations' outcomes were in good agreement with the predicted conditions where cooperation should invade defection, and *vice versa* (Fig. 1.3 and Fig. 1.4). Cooperation level was low in conditions where a population of defectors was predicted to be resistant against the invasion of cooperation (Fig. 1.3, area below the solid line). By contrast, cooperation always went to fixation in conditions where it was predicted to invade a population of defectors and defection was not predicted to invade a population of cooperators (Fig. 1.3, area above the dashed line). Finally, populations were polymorphic in conditions for which both cooperation was predicted to invade a population of defectors and defection was predicted to invade a population of cooperators (Fig. 1.3CDEHJK; TPSM $t = 0.8, 0.5, 0.2$, TUSM $t = 0.8, 0.5, 0.2$, and TSM $s = 3, 5$, intersection of the area below the dashed line and the area above the solid line).

Despite the good agreement between formal analyses and experimental results, there were few small discrepancies. In contrast to the predicted conditions for evolution of cooperation under TUSM (i.e., $r > t$), cooperation evolved when r was slightly lower than t (i.e., $r = 0.78$ instead of 0.8 with TUSM $t = 0.8$, and $r = 0.46, 0.48$ instead of 0.5 with TUSM $t = 0.5$) Similarly, in contrast to the predicted conditions for evolution of cooperation under TSM (i.e., $r > 1/s$), cooperation evolved in conditions when r was slightly lower than $1/s$ ($r = 0.48$ instead of 0.5 with TSM $s = 2$). These small discrepancies stem from the effects of mutation in finite populations, which relaxed the conditions for evolution of cooperation with TUSM and TSM (*Supplementary materials*, Section 1.7.6).

In all eleven treatments and in all conditions, the evolved populations were stable. The only exception was under TPSM with $t = 0.2$ and TUSM with $t = 0.2$ in conditions with r ranging between 0.25 and 0.5 (Fig. 1.5). Due to the low value of the truncation threshold in these two methods of selection, the number of parents was small in relation to the population size. Low effective population size has been shown to suppress selection and amplify random drift (Lieberman et al. 2005), which explains the instability of the evolved populations. Interestingly, high level of relatedness ($r > 0.5$) prevented instability of populations. The combined effect of high relatedness and low truncation threshold on the stability of populations will require further studies.

1.6 Discussion

The results show that cooperation level is positively associated with relatedness level regardless of the selection method used. This supports the long recognized view (Hamilton 1964; Lehmann and Balloux 2007; Waibel et al. 2011) that the relatedness between

individuals or, more generally, the positive assortment between interacting individuals (Eshel and Cavalli-Sforza 1982; Hamilton 1971), facilitates the evolution of cooperation. However, the results also show that selection methods regulate the evolution of cooperation, by strengthening or relaxing the conditions in which cooperation could evolve. For example, a significant level of cooperation evolved in a wide range of conditions under TPSM and TUSM with low truncation threshold ($t = 0.2$), and under TSM with high tournament size ($s = 5$). Since low truncation threshold and high tournament size both reflect strong selection pressure, this suggests that cooperation may originate more easily in harsh environments, or at the beginning of invasion events when the population is not yet well adapted to the new environment.

The level of cooperation was also positively associated with benefits of cooperation on performance, but only under PSM and TPSM. With the other selection methods, the value of the performance benefit did not affect the cooperation level. This is because, RSM, TUSM and TSM select the genomes of the individuals to the descending generation based only on the relative performance scores (i.e., whether one individual has higher performance score than the other). Thus, the effect of cooperation on performance scores does not affect the likelihood of individuals to contribute to the next generation under RSM, TUSM and TSM, provided that the general relationship $B > C > 0$ holds. Consequently, with non-proportionate selection methods all prisoner's dilemma games are equivalent for a given relatedness level, and lead to the same evolutionary outcome. This result is consistent with a previous report on a hawk-dove game played in populations of unrelated individuals (Ficici et al. 2005).

Our analyses also demonstrate that polymorphic populations evolve with TPSM, TUSM, and TSM in some conditions. In polymorphic populations, cooperation and defection co-exist simultaneously which is a sign of frequency-dependent selection. The reproductive advantage of cooperators over defectors depends on their proportion in the population. With low proportion of cooperators, they have the reproductive advantage and increase in numbers. However, with high proportion of cooperators, they lose the reproductive advantage and decrease in numbers. Overall, TPSM, TUSM and TSM act in such conditions as balancing selection that stabilizes the cooperation level at an intermediate value. Similar conclusions were reached for cooperation evolving under selection in ephemeral networks (Godfrey-Smith and Kerr 2009), which in fact resemble much the method of the tournament selection.

It remains an open question which method of selection to use in evolutionary simulations. With PSM, the performance score is equal to fitness since fitness is linearly

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

proportional to performance scores (Appendix A) and because fitness is irrelevant to scaling (Wagner 2010). Thus, evolutionary simulations using PSM directly correspond to mathematical models which usually simply use fitness values (Rice 2004).

Truncation selection was proposed based on the observation that many biological processes display a thresholding effect (Crow and Kimura 1979; Crow et al. 1970; Milkman 1978). For example, only the rabbits with the running speed higher than a certain threshold value may escape predators, survive and reproduce (Crow et al. 1970). Fitness distributions supporting the idea of truncation were observed in social insects in the wild (Rodriguez-Munoz et al. 2010). Overall, truncation selection methods approximate natural selection by predation (Genovart et al. 2010), when the weakest are eliminated from the gene pool. They also resemble to some extent a purifying selection, which removes deleterious mutations (Hurst 2009). From yet another perspective, selection with truncation mimics competition for limited resources in highly mobile species, such as the competition for nest sites in birds.

Finally TSM, which simulates direct competition in small groups, resembles intrasexual selection where individuals of the same sex directly compete between each other to reproduce, and unsuccessful competitors have few or no offspring (Darwin 1859). This method is also similar to selection in ephemeral networks, i.e., short-lasting groups in which individuals interact and compete, which form in microbes, marine invertebrates, annual plants and other organisms (see Godfrey-Smith and Kerr 2009 for more details).

The measurement of fitness of organisms in the wild is difficult, and there has been a considerable discussion about how to measure fitness in natural population and how to represent it mathematically (Wagner 2010). Our formal analyses and simulations indeed show that each of the five commonly used selection methods regulates the evolution of cooperation in a distinct way. The difference in outcomes between the selection methods stems from differences in the mapping between performance and the relative contribution of genotypes to the next generation. The actual mapping between phenotype and fitness poses a great challenge for both evolutionists and ecologists because it likely depends on many factors such as the nature of intra- and interspecific competition (Ferriere and Michod 2011). Consequently, the choice of a selection method, although often marginalized, is a crucial step in the modeling process as it has important implication on the evolutionary outcome of the investigated traits.

1.7 Supplementary materials

1.7.1 Mathematical model of cooperation in populations of related individuals

Cooperative task and relatedness between individuals

We considered an infinite population of individuals which are cooperators or defectors (i.e., non-cooperators). Individuals interact with each other in random pairs. With probability $0 < r < 1$ an individual interacts with itself (e.g., with its identical clone). Otherwise, with probability $1 - r$, the individual interacts with an individual chosen uniformly at random from the population (Eshel and Cavalli-Sforza 1982). Note that r equals to average genetic relatedness between the individuals (Zhang and Hui 2011). In each pair, the two individuals play one iteration of prisoner's dilemma game (Smith and Price 1973) with payoffs R, S, T, P . A cooperator receives a "reward" payoff R when paired with a cooperator, and a "sucker's" payoff S when paired with a defector. A defector receives a "temptation" payoff T when paired with a cooperator, and a "punishment" payoff P when paired with a defector. Cooperation is beneficial ($R > P$) but sensitive to cheating ($T > R$ and $P > S$). The payoffs received by the individuals equal to their performance scores which are translated into fitness by one of the five selection methods: PSM, RSM, TPSM with truncation threshold t , TUSM with truncation threshold t and TSM with tournament size s . For each selection method, we derived the conditions in which cooperation invades a population of defectors, and *vice versa*. Without loss of generality, we assume the payoffs R, S, T, P are non-negative (because PSM and TPSM operate on non-negative performance scores). With PSM and TPSM, we also investigate the linear version of the prisoner's dilemma game with benefit of cooperation $B = R - S = T - P$ and cost of cooperation $C = T - R = P - S$ in its normalized form, i.e., $S = 0$.

Population structure

Let $0 < p < 1$ denote the proportion of cooperators in the population at a given generation. The probability that the given cooperator interacts with a cooperator equals $r + (1 - r)p$, and the probability it interacts with a defector equals $(1 - r)(1 - p)$. The probability that the given defector interacts with a defector equals $r + (1 - r)(1 - p)$, and the probability it interacts with a cooperator equals $(1 - r)p$. Consequently, the proportion of defectors

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

with performance T is equal:

$$p_{dc} = p(1-p)(1-r)$$

The proportion of cooperators with performance R is equal:

$$p_{cc} = p - p(1-p)(1-r)$$

The proportion of defectors with performance P is equal:

$$p_{dd} = (1-p) - p(1-p)(1-r)$$

The proportion of cooperators with performance S is equal:

$$p_{cd} = p(1-p)(1-r)$$

Positive selection of cooperation and of defection

Let P_c denote the probability that a cooperator is chosen by the given selection method to contribute an offspring to the descending generation. Cooperation is under positive selection when there are cooperators in the population:

$$p > 0$$

and the proportion of cooperators in the descending generation is higher than in the current one:

$$P_c/p > 1$$

Let P_d denote the probability that a defector is chosen by the given selection method to contribute an offspring to the descending generation. Defection is under positive selection when there are defectors in the population:

$$1-p > 0$$

and the proportion of defectors in the descending generation is higher than in the current one:

$$P_d/(1-p) > 1$$

1.7.2 Evolution of cooperation under proportionate selection method

Positive selection of cooperation

The probability P_c that PSM chooses a cooperator to contribute an offspring to the descending generation is equal:

$$P_c = \frac{Rp_{cc} + Sp_{cd}}{Rp_{cc} + Sp_{cd} + Tp_{dc} + Pp_{cd}}$$

$$P_c = \frac{R(p - p(1-p)(1-r)) + Sp(1-p)(1-r)}{R(p - p(1-p)(1-r)) + Sp(1-p)(1-r) + Tp(1-p)(1-r) + P(1-p - p(1-p)(1-r))}$$

Cooperation is under positive selection when $p > 0$ and $P_c/p > 1$, which gives:

$$\frac{R(1 - (1-p)(1-r)) + S(1-p)(1-r)}{R(p - p(1-p)(1-r)) + Sp(1-p)(1-r) + Tp(1-p)(1-r) + P(1-p - p(1-p)(1-r))} > 1 \quad (1.1)$$

Defection is near fixation when the invasion of cooperation starts. In the limit of $p \rightarrow 0$, Inequality 1.1 simplifies to:

$$r > \frac{P - S}{R - S} \quad (1.2)$$

Assuming linear and normalized payoffs ($B = R - S = T - P$, $C = T - R = P - S$, $S = 0$), Inequality 1.1 simplifies to:

$$r > \frac{C}{B} \quad (1.3)$$

Note that Inequality 1.2 holds only in the limit of $p \rightarrow 0$, whereas Inequality 1.3 holds for any p such that $0 < p < 1$.

Positive selection of defection

The probability P_d that PSM chooses a defector to contribute an offspring to the descending generation is equal:

$$P_d = \frac{Tp_{dc} + Pp_{cd}}{Rp_{cc} + Sp_{cd} + Tp_{dc} + Pp_{cd}}$$

$$P_d = \frac{Tp(1-p)(1-r) + P(1-p - p(1-p)(1-r))}{R(p - p(1-p)(1-r)) + Sp(1-p)(1-r) + Tp(1-p)(1-r) + P(1-p - p(1-p)(1-r))}$$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

Defection is under positive selection when $1 - p > 0$ and $P_d/(1 - p) > 1$, which gives:

$$\frac{Tp(1-r) + P(1-p(1-r))}{R(p-p(1-p)(1-r)) + Sp(1-p)(1-r) + Tp(1-p)(1-r) + P(1-p-p(1-p)(1-r))} > 1 \quad (1.4)$$

Cooperation is near fixation when the invasion of defection starts. In the limit of $p \rightarrow 1$, Inequality 1.4 simplifies to:

$$r < \frac{T-R}{T-P} \quad (1.5)$$

Assuming linear and normalized payoffs ($B = R - S = T - P$, $C = T - R = P - S$, $S = 0$), Inequality 1.4 simplifies to:

$$r < \frac{C}{B} \quad (1.6)$$

Note that Inequality 1.5 holds only in the limit of $p \rightarrow 1$, whereas Inequality 1.6 holds for any p such that $0 < p < 1$.

1.7.3 Evolution of cooperation under rank selection method

RSM act on ranks of the individuals. Thus, with RSM the derivation follows the one for PSM assuming payoffs equal to appropriate ranks, i.e., $T = 4, R = 3, P = 2, S = 1$. Therefore, cooperation is under positive selection when:

$$r > \frac{1}{2} \quad (1.7)$$

and defection is under positive selection when:

$$r < \frac{1}{2} \quad (1.8)$$

Note that Inequalities 1.7 and 1.8 hold for any p such that $0 < p < 1$, and regardless of the original payoff values.

1.7.4 Evolution of cooperation under truncation selection methods

Truncated selection with truncation coefficient $0 < t < 1$ considers only the t fraction of individuals with the highest performance as viable.

Positive selection of cooperation

Let assume the following:

$$p_{dc} + p_{cc} < t < p_{dc} + p_{cc} + p_{dd} \quad (1.9)$$

Note that $p_{dc} + p_{cc} = p$ and $1 - p < p_{dc} + p_{cc} + p_{dd}$, thus Inequality 1.9 holds when $p < t < 1 - p$. Consequently, Inequality 1.9 holds for a wide range of values of t when p is small. In particular, Inequality 1.9 holds for any $0 < t < 1$ when the invasion of cooperation starts, and defection is near fixation, i.e., $p \rightarrow 0$. When Inequality 1.9 holds, it implies that the set of viable individuals contains all cooperators with performance R , all defectors with performance T and some defectors with performance P .

TPSM t chooses parents from the set of viable individuals at random, proportionally to the performance scores. Therefore, the probability P_c that TPSM t chooses a cooperator to contribute an offspring to the descending generation is equal:

$$P_c = \frac{Rp_{cc} + 0p_{cd}}{Rp_{cc} + 0p_{cd} + Tp_{dc} + P(t - p_{dc} - p_{cc}) + 0(p_{dd} - t + p_{dc} + p_{cc})}$$

$$P_c = \frac{R(p - p(1 - p)(1 - r))}{R(p - p(1 - p)(1 - r)) + Tp(1 - p)(1 - r) + P(t - p)}$$

Cooperation is under positive selection when $p > 0$ and $P_c/p > 1$, which gives:

$$\frac{R(1 - (1 - p)(1 - r))}{R(p - p(1 - p)(1 - r)) + Tp(1 - p)(1 - r) + P(t - p)} > 1 \quad (1.10)$$

Defection is near fixation when the invasion of cooperation starts. In the limit of $p \rightarrow 0$ Inequality 1.10 simplifies to:

$$r > \frac{Pt}{R} \quad (1.11)$$

Assuming linear and normalized payoffs ($B = R - S = T - P$, $C = T - R = P - S$, $S = 0$) Inequality 1.10 simplifies to:

$$r > \frac{Ct - Cp^2}{B + Cp - Bp + Cp^2} \quad (1.12)$$

which in the limit of $p \rightarrow 0$ further simplifies to:

$$r > \frac{Ct}{B} \quad (1.13)$$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

Note that Inequalities 1.11 and 1.13 hold only in the limit of $p \rightarrow 0$, whereas Inequality 1.12 holds for any p such that $0 < p < 1$.

TUSM t chooses parents from the set of viable individuals uniformly at random. Therefore, the probability P_c that TUSM t chooses a cooperator to contribute an offspring to the descending generation is equal:

$$P_c = \frac{1p_{cc} + 0p_{cd}}{1p_{cc} + 0p_{cd} + 1p_{dc} + 1(t - p_{dc} - p_{cc}) + 0(p_{dd} - t + p_{dc} + p_{cc})}$$

$$P_c = \frac{p - p(1-p)(1-r)}{t}$$

Cooperation is under positive selection when $p > 0$ and $P_c/p > 1$, which gives:

$$r > \frac{t-p}{1-p} \quad (1.14)$$

Defection is near fixation when the invasion of cooperation starts. In the limit of $p \rightarrow 0$ Inequality 1.14 simplifies to:

$$r > t \quad (1.15)$$

Note that Inequality 1.15 holds only in the limit of $p \rightarrow 0$, whereas Inequality 1.14 holds for any p such that $0 < p < 1$.

Positive selection of defection

Let assume the following

$$p_{dc} < t < p_{dc} + p_{cc} \quad (1.16)$$

Note that $p_{dc} < 1 - p$ and $p_{dc} + p_{cc} = p$, thus Inequality 1.16 holds when $1 - p < t < p$. Consequently, Inequality 1.16 holds for a wide range of values of t when p is large. In particular, Inequality 1.16 holds for any $0 < t < 1$ when the invasion of defection starts, and cooperation is near fixation, i.e., $p \rightarrow 1$. When Inequality 1.16 holds, it implies that the set of viable individuals contains some cooperators with performance R and all defectors with performance T .

TPSM t chooses parents from the set of viable individuals at random, proportionally to the performance scores. Therefore, the probability P_d that TPSM t chooses a defector to contribute an offspring to the descending generation is equal:

$$P_d = \frac{Tp_{dc} + 0p_{dd}}{R(t - p_{dc}) + 0(p_{cc} - t + p_{dc}) + 0p_{cd} + Tp_{dc} + 0p_{dd}}$$

$$P_d = \frac{Tp(1-p)(1-r)}{R(t - p(1-p)(1-r)) + Tp(1-p)(1-r)}$$

Defection is under positive selection when $1 - p > 0$ and $P_d/(1 - p) > 1$, which gives:

$$\frac{Tp(1-r)}{R(t - p(1-p)(1-r)) + Tp(1-p)(1-r)} > 1 \quad (1.17)$$

Cooperation is near fixation when the invasion of defection starts. In the limit of $p \rightarrow 1$ Inequality 1.17 simplifies to

$$r < 1 - \frac{Rt}{T} \quad (1.18)$$

Assuming linear and normalized payoffs ($B = R - S = T - P$, $C = T - R = P - S$, $S = 0$) Inequality 1.17 simplifies to

$$r < 1 - \frac{Bt}{Bp + Cp^2} \quad (1.19)$$

which in the limit of $p \rightarrow 1$ further simplifies to

$$r < 1 - \frac{Bt}{B + C} \quad (1.20)$$

Note that Inequalities 1.18 and 1.20 hold in the limit of $p \rightarrow 1$, whereas 1.19 holds for any p such that $0 < p < 1$.

TUSM t chooses parents from the set of viable individuals uniformly at random. Therefore, the probability P_d that TUSM t chooses a defector to contribute an offspring to the descending generation is equal:

$$P_d = \frac{1p_{dc} + 0p_{dd}}{1(t - p_{dc}) + 0(p_{cc} - t + p_{dc}) + 0p_{cd} + 1p_{dc} + 0p_{dd}}$$

$$P_d = \frac{p(1-p)(1-r)}{t}$$

Defection is under positive selection when $1 - p > 0$ and $P_d/(1 - p) > 1$, which gives:

$$r < 1 - \frac{t}{p} \quad (1.21)$$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

Cooperation is near fixation when the invasion of defection starts. In the limit of $p \rightarrow 1$ Inequality 1.21 simplifies to:

$$r < 1 - t \quad (1.22)$$

Note that Inequality 1.22 holds only in the limit of $p \rightarrow 1$, whereas Inequality 1.21 holds for any p such that $0 < p < 1$.

1.7.5 Evolution of cooperation under tournament selection method

With TSM the analyses are not as straightforward as with other selection methods. This is because TSM creates “tournaments”, i.e., groups of $s \geq 2$ individuals sampled with replacement uniformly at random from the entire population. The individual with the highest performance score in a tournament is chosen to contribute an offspring to the descending generation.

Positive selection of cooperation

A cooperator with performance R wins a tournament only if not set against defector(s) with performance T . A cooperator with performance S wins a tournament only if set against $s - 1$ cooperators with performance S . Therefore, the probability P_c that TSM chooses a cooperator to contribute an offspring to the descending generation is equal:

$$P_c = \sum_{i=1}^s \binom{s}{i} (p_{cc})^i (p_{dd} + p_{cd})^{s-i} + (p_{cd})^s$$

$$P_c = p_{cc} \sum_{i=1}^s \binom{s}{i} (p_{cc})^{i-1} (p_{dd} + p_{cd})^{s-i} + (p_{cd})^s$$

$$P_c = p_{cc} \left[\binom{s}{1} (p_{cc})^{1-1} (p_{dd} + p_{cd})^{s-1} + \sum_{i=2}^s \binom{s}{i} (p_{cc})^{i-1} (p_{dd} + p_{cd})^{s-i} \right] + (p_{cd})^s$$

$$P_c = p_{cc} \left[s(p_{dd} + p_{cd})^{s-1} + \sum_{i=2}^s \binom{s}{i} (p_{cc})^{i-1} (p_{dd} + p_{cd})^{s-i} \right] + (p_{cd})^s$$

$$P_c = (p - p(1-p)(1-r)) \left[s(1-p)^{s-1} + \sum_{i=2}^s \binom{s}{i} (p - p(1-p)(1-r))^{i-1} (1-p)^{s-i} \right] + (p(1-p)(1-r))^s$$

Cooperation is under positive selection when $P_c/p > 1$, which gives:

$$(1-(1-p)(1-r)) \left[s(1-p)^{s-1} + \sum_{i=2}^s \binom{s}{i} (p-p(1-p)(1-r))^{i-1} (1-p)^{s-i} \right] + p^{s-1} (1-p)^s (1-r)^s > 1 \quad (1.23)$$

The defectors are near fixation when the invasion of cooperations starts. In the limit of $p \rightarrow 0$ Inequality 1.23 simplifies to:

$$(1-(1-0)(1-r)) \left[s(1-0)^{s-1} + \sum_{i=2}^s \binom{s}{i} (0-0(1-0)(1-r))^{i-1} (1-0)^{s-i} \right] + 0^{s-1} (1-0)^s (1-r)^s > 1$$

$$r(s+0) + 0 > 1$$

$$r > \frac{1}{s} \quad (1.24)$$

Note that Inequality 1.24 holds only in the limit of $p \rightarrow 0$, whereas Inequality 1.23 holds for any p such that $0 < p < 1$.

Positive selection of defection

A defector with performance T always wins a tournament. A defector with performance P wins a tournament only if set against defectors with performance P and cooperators with performance S . Therefore, the probability P_d that TSM s chooses a defector to contribute an offspring to the descending generation is equal:

$$P_d = \sum_{i=1}^s \binom{s}{i} (p_{dc})^i (1-p_{dc})^{s-i} + \sum_{i=1}^s \binom{s}{i} (p_{dd})^i (p_{cd})^{s-i}$$

$$P_d = p_{dc} \sum_{i=1}^s \binom{s}{i} (p_{dc})^{i-1} (1-p_{dc})^{s-i} + \sum_{i=0}^s \binom{s}{i} (p_{dd})^i (p_{cd})^{s-i} - (p_{cd})^s$$

$$P_d = p_{dc} \left[\binom{s}{1} (p_{dc})^{1-1} (1-p_{dc})^{s-1} + \sum_{i=2}^s \binom{s}{i} (p_{dc})^{i-1} (1-p_{dc})^{s-i} \right] + (p_{dd} + p_{cd})^s - (p_{cd})^s$$

$$P_d = p_{dc} \left[s(1-p_{dc})^{s-1} + \sum_{i=2}^s \binom{s}{i} (p_{dc})^{i-1} (1-p_{dc})^{s-i} \right] + (p_{dd} + p_{cd})^s - (p_{cd})^s$$

$$P_d = p(1-p)(1-r) \left[s(1-p(1-p)(1-r))^{s-1} + \sum_{i=2}^s \binom{s}{i} (p(1-p)(1-r))^{i-1} (1-p(1-p)(1-r))^{s-i} \right] +$$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

$$+(1-p)^s - (p(1-p)(1-r))^s$$

Defection is under positive selection when $P_d/(1-p) > 1$, which gives:

$$p(1-r) \left[s(1-p(1-p)(1-r))^{s-1} + \sum_{i=2}^s \binom{s}{i} (p(1-p)(1-r))^{i-1} (1-p(1-p)(1-r))^{s-i} \right] +$$

$$+(1-p)^{s-1} - p^s(1-p)^{s-1}(1-r)^s > 1 \quad (1.25)$$

The cooperators are near fixation when the invasion of defection starts. In the limit of $p \rightarrow 1$ Inequality 1.25 simplifies to:

$$1(1-r) \left[s(1-1(1-1)(1-r))^{s-1} + \sum_{i=2}^s \binom{s}{i} (1(1-1)(1-r))^{i-1} (1-1(1-1)(1-r))^{s-i} \right] +$$

$$+(1-1)^{s-1} - 1^s(1-1)^{s-1}(1-r)^s > 1$$

$$(1-r)(s+0) + 0 - 0 > 1$$

$$r < 1 - \frac{1}{s} \quad (1.26)$$

Note that Inequality 1.26 holds only in the limit of $p \rightarrow 1$, whereas Inequality 1.25 holds for any p such that $0 < p < 1$.

1.7.6 Effects of random drift

We have observed small differences between theoretical predictions (Section 1.5.1) and evolutionary simulations (Section 1.5.2). We shall now explain these differences by investigating the effects of random drift caused by mutation, which was present in the simulations and not taken into account in the formal analyses. Note that in simulations we used normalized linear payoffs ($B = R - S = T - P$, $C = T - R = P - S$, $S = 0$) with fixed costs of cooperation ($C = 1$).

Proportionate selection method

Following Inequality 1.3, cooperation is under positive selection when $r > \frac{1}{B}$ for any value of p such that $0 < p < 1$. Consequently, with PSM, random drift does not influence

the conditions for the evolution of cooperation.

Rank selection method

Following Inequality 1.7, cooperation is under positive selection when $r > \frac{1}{2}$ for any value of p such that $0 < p < 1$. Consequently, random drift does not influence the conditions for the evolution of cooperation.

Truncation-proportionate selection method

Following Inequality 1.12, cooperation is under positive selection when $0 < p < t < 1 - p < 1$ and

$$r > \frac{t - p^2}{B + p - Bp + p^2} \quad (1.27)$$

Due to mutations and random drift the value of p may increase by chance. Since Inequality 1.27 depends on p , random drift may influence the conditions for evolution of cooperation. In the limit of $p \rightarrow 0$, Inequality 1.27 simplifies to:

$$r > \frac{t}{B}$$

Assuming p is a small positive fraction (e.g., $0 < p < 0.1$), we may disregard in Inequality 1.27 the components with p in the second power as irrelevant in comparison to components with p in the first and zeroth power, and obtain:

$$\frac{t - p^2}{B + p - Bp + p^2} \approx \frac{t}{B(1 - p) + p} > \frac{t}{B}$$

Consequently, with TPSM, random drift sharpens the conditions for the evolution of cooperation. For example, with $B = 3$, $t = 0.5$ and $p = 0.1$ cooperation invades a population when $r > 0.174$ (instead of $r > 0.167$).

Truncation-uniform selection method

Following Inequality 1.15 cooperation is under positive selection when $0 < p < t < 1 - p < 1$ and

$$r > \frac{t - p}{1 - p}$$

Chapter 1. Selection methods regulate evolution of cooperation in artificial evolution

This in the limit of $p \rightarrow 0$ gives:

$$r > t$$

Consequently, random drift relaxes the conditions for the evolution of cooperation because:

$$t > \frac{t-p}{1-p}$$

For example, with $t = 0.5$ and $p = 0.1$ cooperation invades a population when $r > 0.44$ (instead of $r > 0.5$).

Tournament selection method

Due to the complexity of Inequality 1.23 the analysis of the effects of mutation and random drift is difficult for a general case of TSM with any tournament size $s \geq 2$. Thus, we will perform it solely for $s = 2$. Following Inequality 1.23 we obtain that cooperation invades a population under TSM $s = 2$ when:

$$p^3 r^3 - 2p^3 r + p^3 - 2p^2 r^2 + 4p^2 r - 2p^2 + pr^2 - 4pr + 2p + 2r > 1$$

Assuming p is a small positive fraction (e.g., $0 < p < 0.1$), we may disregard all components with p in the second power or higher as irrelevant in comparison to components with p in the first and zeroth power, and obtain:

$$pr^2 - 4pr + 2p + 2r > 1$$

With $0 < p < 0.5$, the solution for r is:

$$r > \frac{\sqrt{2p^2 - 3p + 1} + 2p - 1}{p}$$

Following Inequality 1.24, the condition for the evolution of cooperation under TSM $s = 2$ in the limit of $p \rightarrow 0$ is

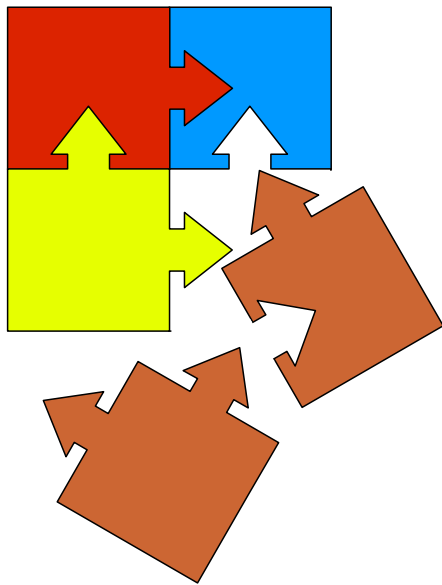
$$r > \frac{1}{2}$$

Consequently, random drift relaxes the conditions for the evolution of cooperation because for $0 < p < 0.5$:

$$\frac{1}{2} > \frac{\sqrt{2p^2 - 3p + 1} + 2p - 1}{p}$$

For example, with $p = 0.1$ cooperation invades a population when $r > 0.48$ (instead of $r > 0.5$).

2 Evolving team compositions by agent swapping



In the previous chapter, we learnt about the prerequisites for division of labor by understanding how selection methods shape the conditions in which cooperation evolves. Here, we take a step further and investigate the evolution of genetically encoded specialization, assuming that agents cooperate and do not cheat. We take an engineering approach and demonstrate how to efficiently and automatically find the optimal fixed assignment of agents to tasks (“team composition”), a problem which often occurs in technical sciences when one designs teams of autonomous agents working together. To this end, we propose a novel algorithm that swaps agents between teams during evolution. This opens the doors to evolve more complex mechanisms of task allocation, in which we will engage in the final chapter. From a methodological standpoint, we demonstrate the importance of crossover operators to maintain a balance between exploration and exploitation aspect of the evolutionary simulation in team optimization.

2.1 Disclosure

Contents of this chapter are taken from an article of the same title, which have been accepted in *IEEE Transactions on Evolutionary Computation* (Lichocki et al. in press). The article is authored by Paweł Lichocki, Steffen Wischmann, Laurent Keller and Dario Floreano. Paweł Lichocki designed the project, implemented and executed simulations, and wrote the manuscript. All authors discussed the results and revised the manuscript. Andrea Maesani, Pavan Ramdya, Krzysztof Krawiec, four anonymous reviewers and two editors contributed useful comments on the manuscript. The work was supported by the Swiss National Science Foundation and an ERC advanced grant.

2.2 Abstract

Optimizing collective behavior in multiagent systems requires algorithms to find not only appropriate individual behaviors but also a suitable composition of agents within a team. Over the last two decades, evolutionary methods have been shown to be a promising approach for the design of agents and their compositions into teams. The choice of a crossover operator that facilitates the evolution of optimal team composition is recognized to be crucial, but so far it has never been thoroughly quantified. Here we highlight the limitations of two different crossover operators that exchange entire agents between teams: restricted agent swapping that exchanges only corresponding agents between teams and free agent swapping that allows an arbitrary exchange of agents. Our results show that restricted agent swapping suffers from premature convergence, whereas free agent swapping entails insufficient convergence. Consequently, in both cases the exploration and exploitation aspects of the evolutionary algorithm are not well balanced resulting in the evolution of suboptimal team compositions. To overcome this problem we propose to combine the two methods. Our approach first applies free agent swapping to explore the search space and then restricted agent swapping to exploit it. This mixed approach turns out to be a much more efficient strategy for the evolution of team compositions compared to either strategy alone. Our results suggest that such a mixed agent swapping algorithm should always be preferred whenever the optimal composition of individuals in a multiagent system is unknown.

2.3 Introduction

The optimization of collective behavior displayed by teams of agents plays a crucial role in an increasing number of applications (Bonabeau et al. 1999, 2000), spanning from software agents (e.g., Mathews et al. 2009; Reynolds 1993; Xiang and Lee 2008) to robotics (e.g., Dorigo et al. 2004; Parker and Zhang 2009; Quinn et al. 2003). Evolutionary computation has been advocated as an effective and promising strategy in this domain (Baldassarre et al. 2003; Nolfi and Floreano 2000). An important question in this respect is the composition of the teams of agents. All agents from one team may either use the same control algorithm (genetically homogenous teams) or employ different ones (genetically heterogenous teams) (Iba 1996; Panait and Luke 2005; Waibel et al. 2009). Evolving homogenous teams does not differ conceptually from evolving single agents, because in both cases only one control algorithm is discovered (Miconi 2003). In contrast, with heterogeneous teams a set of distinct control algorithms must be optimized simultaneously. Consequently, the challenge is not only to find the optimal agents, but also the optimal composition of agents within a team (Waibel et al. 2009). Heterogenous teams are of growing interest in the evolutionary community, because they are expected to perform better than homogeneous teams in problems that require task specialization (Bongard 2000a; Panait and Luke 2005; Waibel et al. 2009).

In heterogeneous team evolution, two genetic encodings can be used: (1) individual encoding, where a genome represents one agent and (2) team encoding, where a genome represents a whole team. Consequently, there are differences in an evolutionary algorithm used with individual and team encoding (Fig. 2.1). With individual encoding, one must decide on a method of grouping agents into temporary teams for purpose of performance evaluation (Fig. 2.1.A). With team encoding, one must choose a crossover operator that exchanges genetic material between teams (i.e., swap agents), and not only between agents (Fig. 2.1.B). Both issues are facets of the same challenge: How to (re)compose agents in teams, in order to facilitate the evolutionary search. This question has already been studied for individual encoding (Table 2.1.A-C) but scarcely addressed for team encoding, although the body of work using team encoding is rich (Table 2.1.D-G).

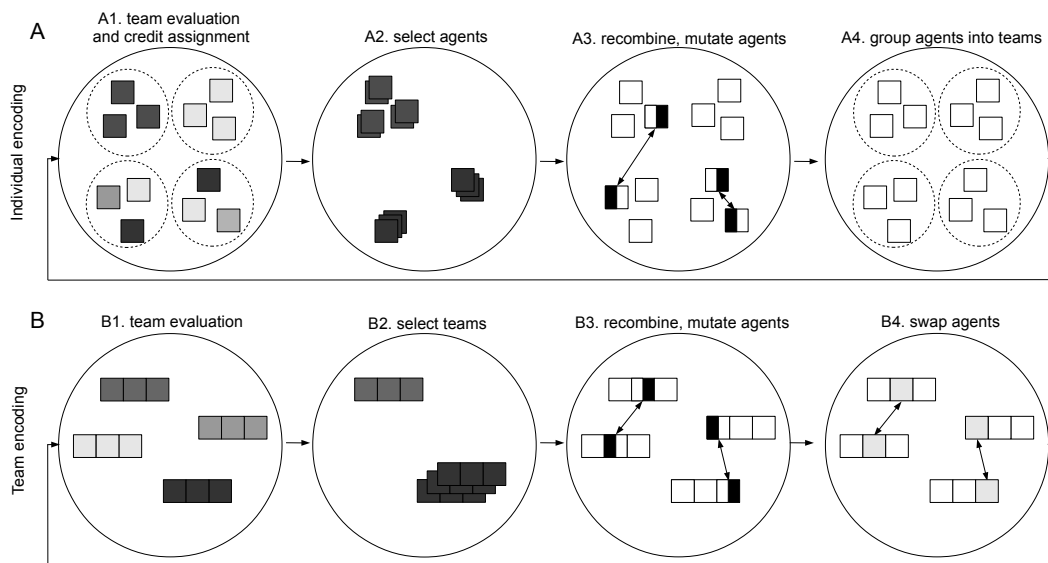


Figure 2.1: The evolutionary algorithm cycle in the evolution of heterogeneous teams using (A) individual encoding and (B) team encoding. (A) Individual encoding: the evolutionary algorithm operates on a population of genotypes, each encoding one agent (depicted as squares). (A1) The agents are grouped into teams and their performance is evaluated together. Each agent must be assigned with a fitness value (the credit assignment problem, see Grefenstette 1988; Panait and Luke 2005; Waibel et al. 2009). A straightforward way of addressing this problem is to distribute the team's performance score equally among the team members (top two teams in A1). This is known as global reward (Panait and Luke 2005), or team-level selection (Waibel et al. 2009). Alternatively, the agents can be assigned with different fitness values, proportional to their personal contribution into the team's performance score (bottom two teams in A1). This is known as local reward (Panait and Luke 2005), or individual-level selection (Waibel et al. 2009). For simplicity, global and local rewards are illustrated together in one population, but typically only one kind of reward is used with all teams from the population. Next, (A2) the algorithm proceeds with the selection of the agents accordingly to their fitness values. In (A1) a darker color denotes higher fitness values, which translates into more copies of fitter agents (A2). Afterwards, (A3) the agents' genotypes are recombined or mutated. Finally, (A4) the agents are grouped into new teams before the performance evaluation and the algorithm starts over with (A1). (B) Team encoding: the evolutionary algorithm operates on a population of genotypes, each encoding one entire team (a square depicts a part of the team's genome corresponding to one agent). Consequently, there is no need for an extra step of grouping agents into teams and the algorithm begins at once with (B1) the evaluation of the teams' performance. There is no credit assignment problem, because team encoding implies that all agents in the team share the same fitness. Next, (B2) the teams are selected according to their fitness values. In (B1) a darker color denotes higher fitness values, which translates into more copies of fitter teams (B2). Then, (B3) the team's genotypes are subject to mutation or agent recombination. Finally, (B4) entire agents may be swapped between the teams. This concludes the cycle and the algorithm starts over with (B1). Note that swapping agents between teams in team encoding conceptually corresponds to grouping of agents into teams in individual encoding.

Table 2.1: A sample of approaches in the evolution of heterogeneous teams using individual encoding and team encoding.

Individual encoding	
Agent grouping	References
(A) One team	Miconi (2001), Werner and Dyer (1991), Ficici et al. (1999), Stanley et al. (2005), Watson et al. (2002), Spector et al. (2005), Ward et al. (2001), Simoes and Barone (2002), Bianco and Nolfi (2004), Agah and Bekey (1997), Cangelosi and Parisi (1998)
(B) Many teams	Waibel et al. (2009), Quinn (2001), Suzuki and Arita (2006), Agah and Tanie (1997), Bull (1998), Eiben et al. (2007), Floreano et al. (2007), Soule and Heckendorn (2008)
(C) Sub-populations	Iba (1996), Eiben et al. (2007), Potter and De Jong (1994), Potter and De Jong (2000), Wiegand et al. (2001), Mirolli and Parisi (2005), Panait et al. (2006)
Team encoding	
Crossover	References
(D) Restricted agent recombination	Iba (1996), Miconi (2003), Bongard (2000a), Miconi (2001), Suzuki and Arita (2006), Soule and Heckendorn (2008), Andre and Teller (1999), Brameier and Banzhaf (2001), Hara and Nagao (1999), Luke et al. (1998), Luke and Spector (1996), Botee and Bonabeau (1998), Haynes and Sen (1996), Haynes and Sen (1997)
(E) Free agent recombination	Suzuki and Arita (2006), Brameier and Banzhaf (2001), Luke and Spector (1996), Haynes and Sen (1997), Agogino et al. (2000)
(F) Restricted agent swapping	Andre and Teller (1999), Luke et al. (1998)
(G) Free agent swapping	-

In team encoding a single genotype encodes the entire team, which makes it decomposable into parts corresponding to the agents. Thus, a crossover operator may exchange genetic material on two levels (Miconi 2003). First, the crossover can recombine the genetic material between agents from the parenting teams. We refer to this process as agent recombination (Table 2.1.DE and Fig. 2.2.AB). Second, the crossover can swap entire agents between the parenting teams. We refer to this process as agent swapping (Table 2.1FG and Fig. 2.2.CD). In contrast to agent recombination, agent swapping does not exchange genetic material between the agents. Consequently, the purpose of agent

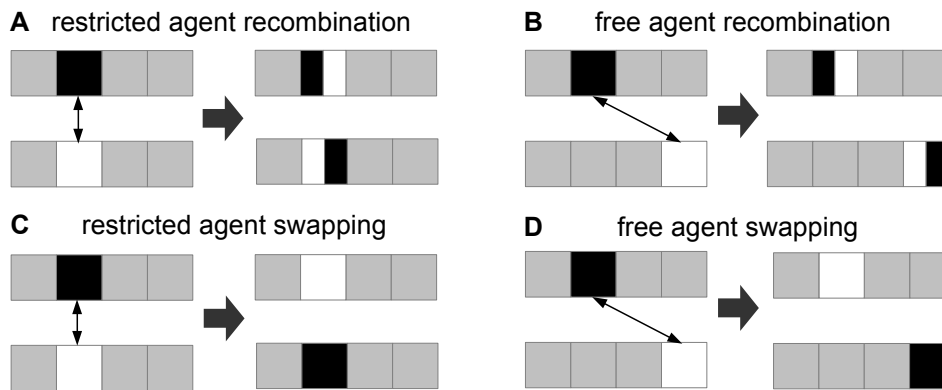


Figure 2.2: Applying four crossover operators on the genotype of two teams of four agents each. Black and white segments of teams' genotypes correspond to agents taking part in the crossover. Grey segments of teams' genotypes correspond to agents not taking part in the crossover. The crossover can either recombine the genetic material from two agents from parenting teams (A and B), or swap entire agents between parenting teams (C and D). In addition, the crossover may be restricted to act only on agents on corresponding positions in parenting teams (A and C). Or, it may be free to act on any agents from parenting teams (B and D).

recombination is to discover "good" agents, and the purpose of agent swapping is to discover "good" team compositions. In addition, one may consider a team to be an ordered sequence of agents. In such a case, the crossover may be restricted to act only on agents on corresponding positions in the parenting teams (Table 2.1.DF and Fig. 2.2.AC). Or, it may be free to act on any agents from the parenting teams (Table 2.1.EG and Fig. 2.2.BD).

In contrast to agent recombination (see, e.g., Haynes and Sen 1997; Luke and Spector 1996; Suzuki and Arita 2006), no attempts have been made to quantify the efficiency of agent swapping in the evolution of teams (Panait and Luke 2005). In particular, it has not been tested if and why using agent swapping leads, or not, to the evolution of optimal team compositions. Here, we experimentally compare restricted agent swapping (RAS) and free agent swapping (FAS) in a problem of finding the optimal team composition. We consider multiple agents that need to divide the labor in order to achieve top performance, i.e., the optimal team is composed of distinct groups of genetically identical agents. We focus on team encoding, which assumes team level of selection (Fig. 2.1) that has been advocated as an efficient strategy in the optimization of teams (Waibel et al. 2009). Consequently, agent interactions and their impact on individual selection pressures are out of scope of this study. Also, we focus on the

evolution of team compositions, and not on the evolution of agents themselves. Thus, we mainly consider large teams of agents having small genomes, i.e., a valid single agent is relatively easy to evolve. We further elaborate on these assumptions in Section 2.6.

We highlight the limitations of both RAS and FAS and explain the conditions under which they fail to evolve teams displaying the optimal composition. These limitations are opposite to each other. With RAS the evolutionary algorithm suffers from premature convergence of the population, whereas with FAS it suffers from insufficient convergence of the population. Consequently, in both cases the exploration and exploitation aspects of the evolutionary algorithm are not well balanced. To overcome this problem we propose to combine the two methods, i.e., first use FAS to explore the search space, and then use RAS to exploit it. This mixed approach proves to be a more efficient strategy in the evolution of team compositions than restricted or free agent swapping alone. Finally, we also validated RAS and FAS on a problem of optimizing decentralized controllers for task allocation and discussed our results in the context of other real-life applications.

2.3.1 Background

The differentiation into individual and team encoding resembles an old discussion on evolving rule-based systems (De Jong 1988). In the approach taken by De Jong, dubbed “the Pitt approach”, a single individual encoded the entire rule set. In contrast, in the approach taken by Holland, dubbed “the Michigan approach”, a single individual encoded just a single decision rule, and the entire population corresponded to the rule set (see De Jong (1988) and references therein for more details).

The approaches using individual encoding can be classified into three categories, according to how the agents are grouped into teams for the purpose of the performance evaluation. With the “one team” approach, all agents from the population are evaluated together, i.e., they compose one team (Table 2.1.A). This method is often used with a continuously updated gene-pool (“steady-state evolution”). With the “many teams” approach, the agents are randomly grouped into many teams (Table 2.1.B). This method is often used with separate gene-pools for subsequent generations (“generational evolution”). With the “sub-populations” approach there are separate subpopulations of agents (Table 2.1.C). An individual is evaluated by teaming it up with individuals from other subpopulations. This method is known as cooperative co-evolution.

In individual encoding the teams are created ad hoc for the purpose of performance evaluation. Consequently, there are no genetic operators applied at the team level. In

contrast, in team encoding we distinguish four qualitatively different types of crossover operators that exchange genetic material between the teams (Fig. 2.2). The crossover either recombines the agents from the parenting teams (Table 2.1.DE) or it swaps entire agents between the parenting teams (Table 2.1.FG). In addition, the crossover is either restricted to act only on agents on corresponding positions in the parenting teams (Table 2.1.DF), or it is free to act on any agents from the parenting teams (Table 2.1.EG). Note that agent swapping is in fact a special case of agent recombination, where the recombination points are always chosen at the beginning (or at the end) of the two genotypes' parts that encode the parenting agents.

The concept of restricted (also called "fixed" by Suzuki and Arita (2006)) and free (also called "unfixed" by Suzuki and Arita 2006 and "interpositional" by Brameier and Banzhaf 2001) agent recombination was introduced independently by Luke and Spector (1996) and by Haynes and Sen (1996). Agent recombination was called "inner crossover" by Miconi (2003). Restricted agent swapping (RAS), called "team transformation" by Andre and Teller (1999), was introduced by Luke et al. (1998) and by Andre and Teller (1999). Free agent swapping (FAS) has not been studied directly yet (Table 2.1.G). Note that evolutionary algorithms with individual encoding, random grouping of agents into many teams and global reward yield high resemblance to team encoding with free agent swapping (Fig. 2.1). This includes some of the work referenced in Table 2.1.B (i.e., Bull 1998; Eiben et al. 2007; Floreano et al. 2007; Soule and Heckendorn 2008; Suzuki and Arita 2006; Waibel et al. 2009).

Haynes and Sen (1996) noted that RAS may be implemented as uniform crossover (i.e., swapping bits on corresponding positions between two genotypes with some probability), with the difference that instead of bits it swaps entire agents. Here, we analogously note that FAS may be implemented as shuffle-uniform crossover (i.e., swapping bits on shuffled positions between two genotypes with some probability), with the difference that instead of bits it swaps entire agents. The family of uniform crossovers was introduced by Syswerda (1989) and analytically studied by Eshelman et al. (1989) and De Jong and Spears (1992), but not in the context of team evolution. In addition, Miconi (2003) and Nakashima et al. (2006) used a simple 1-point or 2-point crossover to recombine the teams' genotypes. This approach leans itself to the category of restricted crossovers. During the crossover, it usually swaps entire agents between teams, with the exception of the agents that happen to be placed on the crossover cutting points. The 1-point and 2-point crossovers, and generally n -point crossover, have been shown to have less exploratory power than uniform crossover (Eshelman et al. 1989).

A few attempts have been made to quantify the efficiency of agent recombination for the evolution of teams, but the studies are inconclusive (Panait and Luke 2005). Some authors advocate using restricted agent recombination (Haynes and Sen 1997; Luke and Spector 1996) and some are proponents of free agent recombination (Brameier and Banzhaf 2001). The problem of evolving team compositions has been addressed only for genetic programming by Hara and Nagao (1999) (automatically defined groups) and by Bongard (2000a) (the Legion system). In these two approaches both the agents and their partitioning into separate sub-teams are evolved together. These approaches are promising, but designed for the tree representation typical for genetic programming. In addition, authors considered only a small number of evolving sub-teams (maximum number of sub-teams: 6 in Hara and Nagao 1999 and 3 in Bongard 2000a). Importantly, there is no study that quantifies the efficiency of agent swapping for the evolution of teams.

Operators similar to RAS and FAS have been proposed in different fields of evolutionary computation. Agent swapping is equivalent to root crossover in genetic programming, which swaps whole trees instead of subtrees (Luke et al. 1998). Also in genetic programming, Koza (1995) studied operators altering the architecture of a multi-part program, which he used with automatic defined functions. These operators included branch duplication and deletion, which in the context of evolving team composition would translate into agent deletion and duplication, respectively. In gene expression programming, Ferreira (2001) proposed gene recombination and gene transposition. Gene recombination swaps entire genes between the parenting genotypes, and thus is equivalent to restricted agent swapping. Gene transposition overwrites one gene with a copy of another gene, and thus is equivalent to agent deletion and duplication performed jointly. Finally, in the field of evolutionary strategies, Sebag and Schoenauer (1996) proposed mutation by imitation. With imitation the probability of an allele's mutation depends on whether the allele is the same or different as the corresponding alleles in some of the best/worst individuals in the population. Consequently, mutation by imitation might be considered a sort of restricted recombination.

2.4 Methods

2.4.1 Restricted and free agent swapping

We evolve teams of agents using team encoding and study restricted agent swapping (RAS, Fig. 2.3.A) and free agent swapping (FAS, Fig. 2.3.B). We implement RAS as

Chapter 2. Evolving team compositions by agent swapping

a uniform crossover (Syswerda 1989), which exchanges the i th agent from the first team with the i th agent from the second team, with probability p for each agent. In an additional experiment we tested RAS implemented as a 1-point and 2-point crossover (*Supplementary materials*, Section 2.7.1).

We implement FAS as a shuffle-uniform crossover (Syswerda 1989), which exchanges the i th agent from the first team with the $S(i)$ th agent from the second team, with probability p for each agent. S is a random permutation of integers from 1 to M , where M is the number of agents in the team. RAS may be considered a specific case of FAS where the sequence S is set to $1, 2, \dots, M$. For sake of simplicity we consider a situation with only two parenting teams, but the operators can scale to any number of parenting teams.

In addition, we theoretically investigate the connections between RAS, FAS and other operators that alter the team composition inspired by architecture-changing operators (Koza 1995) used in genetic programming and gene expression programming (*Supplementary materials*, Section 2.7.7).

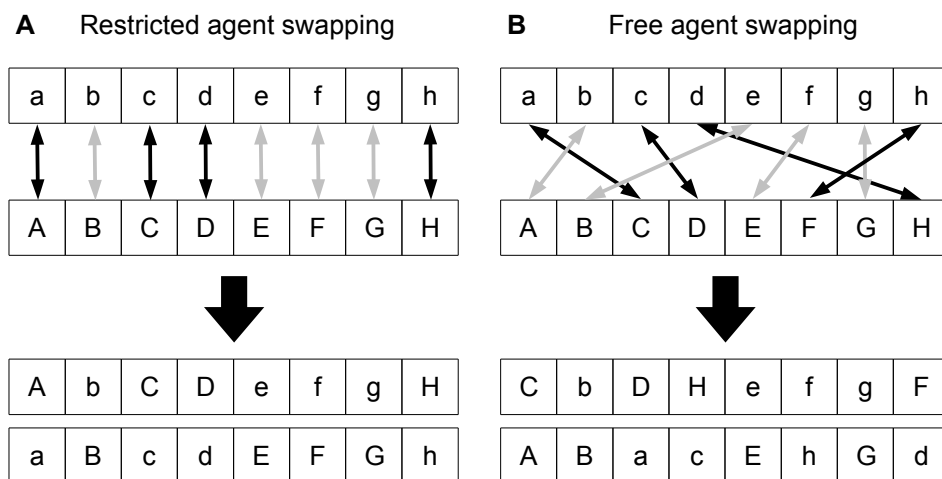


Figure 2.3: Restricted agent swapping (A) and free agent swapping (B) applied on two teams of eight agents. In the presented example the probability of agent swapping was $p = 0.5$, black lines denote the exchanged agents and the grey lines denote the agents that happen to be kept in their original team. The random permutation used to reshuffle the agents before the swap in (b) is $(3, 1, 4, 8, 2, 5, 7, 6)$.

2.4.2 Problem formulation

We experimentally compare RAS and FAS in the evolution of agent teams facing a problem of finding the optimal team composition (i.e., the division of a team into groups of identical agents). Our aim is to mimic a situation when a team needs to display a certain composition of agents in order to achieve top performance (Berman et al. 2009). For example, multiple robots may be more efficient if they compose distinct groups, each focusing on a different task (Dahl et al. 2009). In biology, this process is known as division of labor (see Beshers and Fewell 2001; Duarte et al. 2011; Smith et al. 2008b for comprehensive reviews). For example, in many species of honey bees and ants, some workers forage for food, others care for the brood, others perform maintenance work in the nest, etc. (Fewell 2003; Gordon 1996; Robinson et al. 2009; Robinson 1992).

We consider a team consisting of M agents. There are D distinct types of agents. Thus, there are overall D^M different teams possible. We assume that the optimal team contains K distinct agents, each repeating R times (for simplicity we assume $M = K \cdot R$). Thus, the optimal composition of a team is defined as “ K groups of R agents”, where the agents between the groups are different and the agents within a group are identical. The optimal team is homogeneous for $K = 1$ and $R = M$. The optimal team is heterogeneous for $K = M$ and $R = 1$. And, the optimal team is hybrid (Panait and Luke 2005) (also called partially heterogeneous, see, e.g., Waibel et al. 2009) for $1 < K < M$ and $1 < R < M$.

For a real-life problem the optimal team composition, and consequently the values of K and R are unknown. They are discovered by means of artificial evolution, driven by a fitness function F that measures team performance for a given problem. It should be expected that team performance is correlated with the composition of the team. Thus, we set K and R *a priori* and we define a fitness function f that directly depends on the proportion of “proper” agents in each of the K groups $f = \sum_{j=1}^K \min(\frac{R}{M}, x_j)$. The value x_j is the fraction of agents from a team that belong to the j th group and operator $\min(a, b)$ takes a value a if $a < b$, and b otherwise.

For example, consider three types of agents: A , B and C . Let (x, y, z) denote the number of agents A , B and C , respectively, in a team. The team size is set to six (i.e., $x + y + z = 6$), and the optimal team consists of three agents A and three agents B , i.e., $(3, 3, 0)$. Exactly one team $(0, 0, 6)$ has the lowest performance $f = 0$. Exactly one team $(3, 3, 0)$ has the maximal performance $f = 1$. And, for instance, teams $(1, 1, 4)$, $(0, 2, 4)$ and $(2, 0, 4)$ all have performance $f = 0.33$; teams $(6, 0, 0)$, $(3, 0, 3)$ and $(1, 2, 3)$ all have performance $f = 0.5$; and teams $(2, 2, 2)$, $(1, 5, 0)$ and $(1, 3, 2)$ all have performance $f = 0.67$.

The fitness function f , although not directly applicable to real-life problems, lets us to test the efficiency of RAS and FAS in the evolution of team composition under controlled conditions. The advantage of our approach is the ease of generating instances of various complexities and sizes. This enables us to perform systematic studies and thus draw statistically significant conclusions. We believe that the presented formulation abstracts well the core properties of many problems when one needs to optimize the team composition in a multi-agent system. We further elaborate on this issue in Section 2.6. In addition, we show the applicability of our results by validating the efficiency of RAS and FAS in the evolution of decentralized controllers in a task-allocation problem (Appendix B).

2.4.3 Evolutionary experiments

We evolve teams of agents in three treatments: 1) using RAS for all generations of the evolutionary algorithm, 2) using FAS for all generations of the evolutionary algorithm and 3) using FAS for the first half, and RAS for the second half, of all generations of the evolutionary algorithm. We also investigated alternative ways of combining FAS and RAS (*Supplementary materials*, Section 2.7.6).

We compare RAS and FAS under three conditions, where the optimal team is composed of 1000, 100 and 10 groups of 1, 10 and 100 identical agents, respectively ($K \in \{1000, 100, 10\}$ and $R \in \{1, 10, 100\}$). Overall, this makes a total of 9 experimental lines (3 treatments \times 3 conditions). Each experimental line is replicated 10 times. In all numerical experiments we use populations of 1000 teams of $M = 1000$ agents each. Population size and team sizes are kept constant across generations. The number of all distinct types of agents is set to $D = 10000$. Each evolutionary run lasts for 1000 generations, with the exception of an additional experiment, where the number of generations is set to 2000 (see Appendix B for more details).

The software testbed has been implemented with the help of ECJ framework (Luke 2010). The numerical experiments have been run on the Pleiades cluster at EPFL, Lausanne.

2.4.4 Genetic architecture, selection and reproduction

A team's genotype consists of 1000 alleles (one allele per agent), which are integers from 1 to 10000. At the first generation of each evolutionary run, each of 1000×1000 alleles is independently set to a random integer value between 1 and 10000 with uniform

distribution. Teams are evaluated in the collaborative task (i.e., how similar the team's composition and the optimal composition are) and assigned a performance (see Section 2.4.2). To construct the 1000 teams of the following generation we select 500 times two teams. Each parent is independently selected from the current population using tournament selection with tournament size set to 2, with the exception of an additional experiment, where tournament size is set to 7 (*Supplementary materials*, Section 2.7.4). The two selected teams are reorganized with FAS or RAS, which results in two new teams that are added to the next generation's population. We use RAS and FAS with the probability of exchanging the agents between two teams set to $p = 0.5$ for each position in a team, with the exception of an additional experiment, where $p = 0.2$ is used (*Supplementary materials*, Section 2.7.2). Note that $p = 0.5$ is the highest possible value, because swapping agents between teams A and B with probability $0.5 < p \leq 1$ is equivalent to swapping agents between teams B and A with probability $1 - p$. The newly added teams are not subject to mutation, with the exception of an additional experiment, where each allele is independently set to a random integer value between 1 and 10000 (uniform distribution) with a probability 0.001 (*Supplementary materials*, Section 2.7.3). We do not use mutations in the main experiments for two reasons. First, our intention is to investigate the evolutionary dynamics of RAS and FAS. Thus, to get clearer results on the effects of agent swapping, we do not use the mutation. Second, with 10000 different agents, the population of 1000×1000 agents already contains each agent on average 100 times. Thus, the introduction of the innovative genetic material during evolution should not be necessary, if the agent swapping does efficiently compose optimal teams. Finally, one of the 1000 new teams of the following generation is randomly chosen, discarded and replaced by an exact copy of the best team from the current generation (i.e., elitism of size 1).

2.4.5 Statistical analysis

To compare the teams evolved with RAS and FAS we average, for each experimental line and replicate, team performance over 1000 teams (10 replicates) at generation 1000. We report also the best team performance from 1000 teams (10 replicates) (*Supplementary materials*, Section 2.7.5). We explain the differences in team performance in terms of variation between teams that RAS and FAS introduced into the population. To this aim, we calculated the standard deviation of teams' performance in a population for each of the 1000 generations. In particular, we compared the convergence time (i.e., the number of generations until the measured standard deviation reached zero) and the final variation level between the teams (i.e., the value of the measured standard

deviation at generation 1000). To explain the differences in variation between teams, we compared the proportion of corresponding positions that in all teams contained only agents that could not become members of the optimal team. And, we compared the proportion of agents in teams that could become members of the optimal team, averaged over 1000 teams. The last two characteristics were calculated every 25 generations, due to high demand for resources of these calculations (i.e., CPU and disk space). Statistical significance within multiple experimental lines was determined with Kruskal-Wallis test (nonparametric one-way analysis of variance). Statistical significance between a pair of experimental lines was determined with Wilcoxon test (rank sum test for equal medians).

2.5 Results

We analyze the first two treatments (RAS and FAS) in Sections 2.5.1 and 2.5.2, and the third treatment (mixed FAS/RAS) in Section 2.5.3.

2.5.1 Restricted and free agent swapping

There were important differences in team performance between the three conditions (1000, 100 and 10 groups) at the 1000th generation for both treatments (RAS: Fig. 2.4.A, Kruskal-Wallis test, $df = 2$, $p < 0.001$; FAS: Fig. 2.4.B, Kruskal-Wallis test, $df = 2$, $p < 0.001$). With RAS, the highest team performance was for 1000 groups (100 groups: -27.4% ; 10 groups: -69.8% ; three pairwise Wilcoxon tests $df = 9$, $p < 0.001$). In contrast, with FAS the highest team performance was for 10 groups (100 groups: -9% ; 1000 groups: -35.4% ; three pairwise Wilcoxon tests $df = 9$, $p < 0.001$).

The performance difference between conditions in treatments with RAS and FAS was caused by convergence issues, different for each of the two treatments. With RAS, the performance difference was associated with the convergence time, i.e., the number of generations until the standard deviation of team performance in the population reached zero (Fig. 2.5.A, mean \pm s.d. generations for 1000 groups: 334.3 ± 14.6 ; 100 groups: 294.2 ± 36.7 ; 10 groups: 113.5 ± 3.1 ; Kruskal-Wallis test, $df = 2$, $p < 0.001$; three pairwise Wilcoxon tests $df = 9$, $p < 0.01$). Therefore, the evolutionary algorithm using RAS suffered from “premature convergence”. In contrast, with FAS the performance difference was associated with the variation between the evolved teams, i.e., the value of the standard deviation of team performance in the population at generation 1000 (Fig. 2.5.B; Kruskal-Wallis test, $df = 2$, $p < 0.001$; three pairwise Wilcoxon tests $df = 9$,

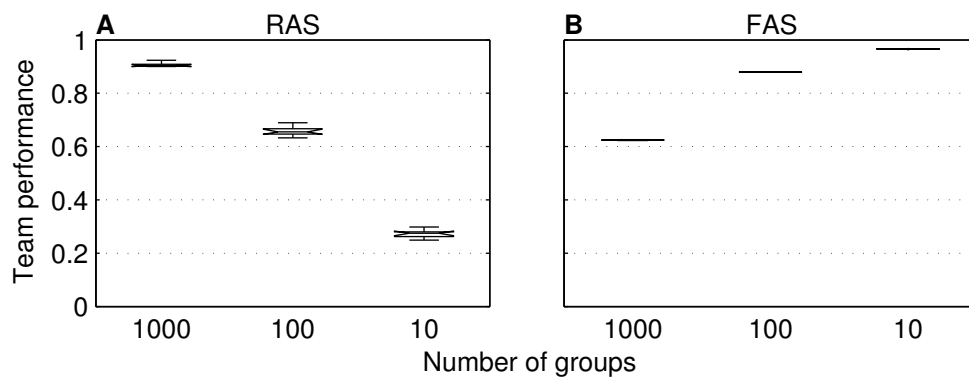


Figure 2.4: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (1000, 100 and 10 groups of identical agents) in order for the team to achieve the optimal performance.

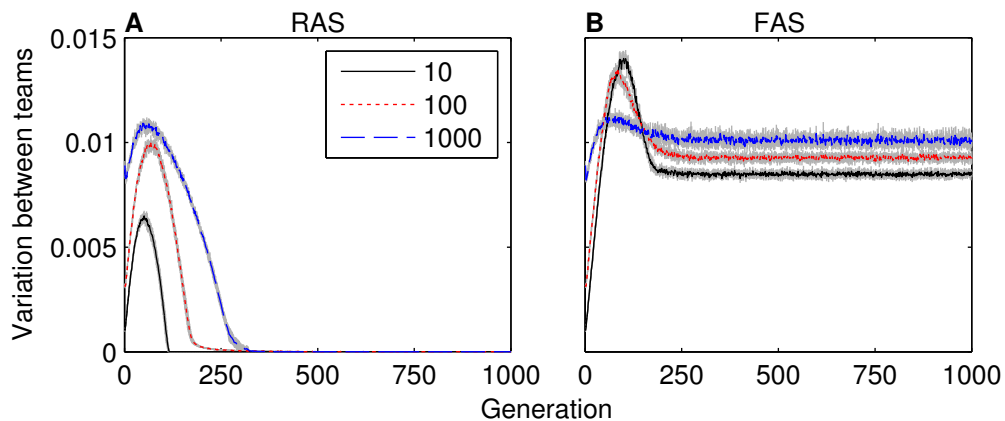


Figure 2.5: Mean variation (\pm s.d. in grey) between the 1000 teams in a population (10 replicates), measured for all 1000 generations. The variation between the teams was quantified with the standard deviation of teams' performance in a population for (A) restricted agent swapping and (B) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition).

$p < 0.001$). Therefore, the evolutionary algorithm using FAS suffered from “insufficient convergence”.

With RAS, premature convergence was detrimental to team performance, because it lead to the disappearance of agents required in the optimal team on corresponding positions in teams across the entire population (Fig. 2.6.A). This was not the case for treatments with FAS (Fig. 2.6.B). Consequently, in treatments with RAS the teams contained agents that could not become members of the optimal team (Fig. 2.6.C). With FAS, insufficient convergence was detrimental to team performance, because it

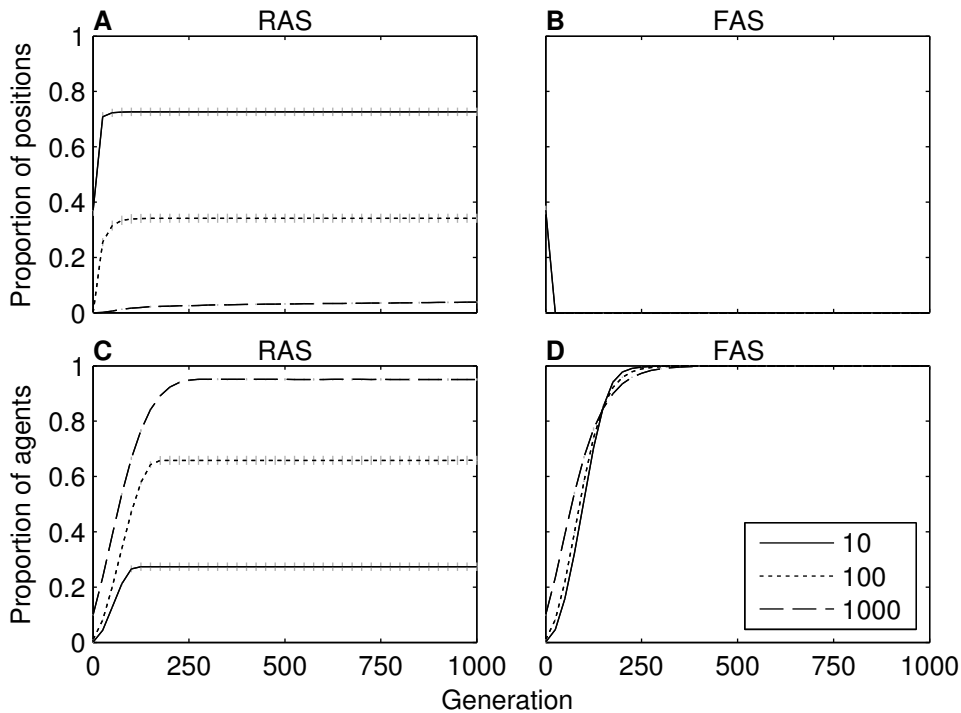


Figure 2.6: A and B: mean proportion (\pm s.d. in grey) of corresponding positions that contained in all 1000 teams only the agents that could not become members of the optimal team. C and D: mean proportion (\pm s.d. in grey) of agents in a team that could become members of the optimal team, averaged over 1000 teams. The populations were analyzed every 25 generations over 10 replicates for (A and C) restricted agent swapping and (B and D) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition).

prevented the formation of the optimal composition. This is supported by the fact that team performance stayed low, even though the teams contained only the agents required in the optimal team (Fig. 2.6.D). We illustrate this process with a simple thought experiment. Consider a population consisting of two identical teams each containing agents A and B (in that order). In such conditions RAS may swap agent A with agent A only, and agent B with agent B only. This does not change team compositions in the population, and thus does not affect the teams' performance. In contrast, FAS may at some point swap agent A with B, which would result in one team containing both agents A, and the second team containing both agents B. Consequently, FAS may destroy favorable team compositions, even when entire population contains only optimal teams.

We performed a sensitivity analysis to see how our results were affected by: lower probability of agent swapping, by mutations and by higher selection pressure. Lower probability of agent swapping had a small detrimental effect on team performance in

both treatments, and was more marked for RAS (*Supplementary materials*, Section 2.7.2). In treatments with RAS, using the mutation counterbalanced to some extent premature convergence (compare Fig. 2.5 and Fig. 2.17), which lead to an increase in team performance. In contrast, with FAS, the mutation introduced additional undesirable variation between the teams, and thus had a weak detrimental effect on the performance (*Supplementary materials*, Section 2.7.3). In treatments with RAS, stronger selection increased the detrimental effect of premature convergence on team performance. With FAS, stronger selection did not overcome the problem of insufficient convergence and had no effect on the performance of the evolved teams (*Supplementary materials*, Section 2.7.4).

We tested alternative implementations of RAS, i.e., 1-point and 2-point crossover. The results indicate that uniform crossover is better suited for restricted agent swapping than n -point crossover (*Supplementary materials*, Section 2.7.1). Here, we presented the mean team performance in the population (Fig. 2.4). We report also the best team performance in the population (averaged over 10 replicates), which shows not to be qualitatively different from the mean team performance (*Supplementary materials*, Section 2.7.5).

2.5.2 Restricted and free agent swapping - disparities in the initial population

In the previous experiments, the agents in the teams were initialized randomly with uniform distribution (i.e., each initialized agent was set to be one of the 10000 possible agents with probability 0.0001). Consequently, the proportion of agents in the initial population that could become members of the optimal team varied between conditions. In the first condition with 1000 groups, the initial population contained on average 10% of agents that could become members of the optimal team. In the second condition with 100 groups, the initial population contained 1% of such agents. And in the third condition with 10 groups, the initial population contained only 0.1% of such agents. We tested if these disparities in the initial population had influence on the difference in team performance between conditions for both treatments. To this aim, we performed a control experiment in which the populations were initialized randomly with a bias. For each of the three conditions, the initial population always contained on average 10% of agents that could become members of the optimal team.

In treatments with RAS and without disparities in the initial population, there was an important increase in team performance for conditions with 10 and 100 groups in the optimal composition (compare Fig. 2.4.A and Fig. 2.7.A; two Wilcoxon tests, $df = 9$,

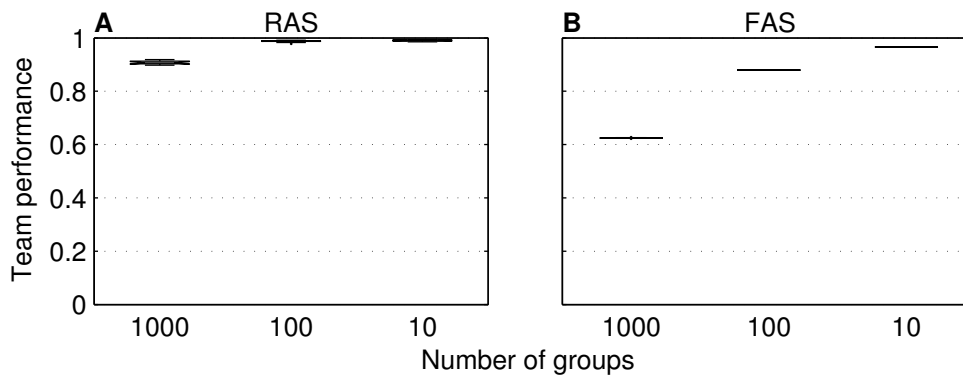


Figure 2.7: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance. The populations were initialized randomly with a bias. For each of the three conditions and for both treatments, the initial population always contained on average 10% of agents that could become members of the optimal team.

$p < 0.001$). By contrast, there was no significant change over the 1000 generations for 1000 groups (compare Fig. 2.4.A and Fig. 2.7.A; Wilcoxon test $df = 9$, $p = 0.47$). This suggests that with RAS team performance primarily depended on the number of agents that could become members of the optimal team. The performance of the evolved teams was higher with more such agents in the initial population.

In treatments with FAS, in contrast to RAS, there were no important differences in performance of teams evolved with and without disparities in the initial population for each of the three conditions (compare Fig. 2.4.B and Fig. 2.7.B, three pairwise Wilcoxon tests $df = 9$, $p > 0.3$). This suggests that with FAS team performance primarily depended on the optimal composition. The performance of the evolved teams was higher with a lower number of groups in the optimal team.

2.5.3 Combining free and restricted agent swapping

In the treatments with RAS and FAS, the exploration and exploitation aspects of the evolutionary algorithm were not well balanced. With RAS the evolutionary algorithm suffered from premature convergence, whereas with FAS it suffered from insufficient convergence. Consequently, in both cases only suboptimal solutions have evolved (Fig. 2.8.AB). In order to overcome the limitations of both RAS and FAS, we combined the two swapping methods in a complementary way: for the first 500 generations FAS was used, and for the next 500 generations RAS was used. We expected this would allow to

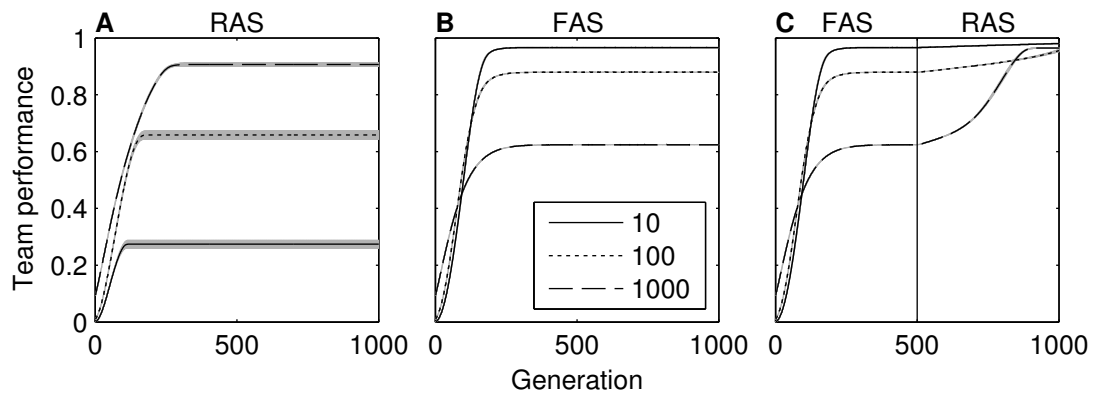


Figure 2.8: Mean performance (\pm s.d. in grey) of 1000 teams evolved with (A) restricted agent swapping, (B) free agent swapping and (C) mixed FAS/RAS (10 replicates). In (c) the teams were evolved with free agent swapping in generations from 1 to 500 (inclusive), and with restricted agent swapping in generations from 501 to 1000. The black, solid vertical line at generation 500 marks the transition from using FAS to using RAS. Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance.

efficiently explore the search space first and then to exploit it. We tested the efficiency of the mixed approach in the evolution of team composition for the three conditions (1000, 100 and 10 groups in the optimal composition).

There were important differences in team performance between the treatments (RAS, FAS, FAS/RAS) at the 1000th generation for each of the three conditions (Fig. 2.8, three Kruskal-Wallis tests, $df = 2$, $p < 0.001$). For each of the three conditions, the performance was higher with FAS/RAS than with RAS alone (Fig. 2.8.AC, three Wilcoxon tests $df = 9$, $p < 0.001$) and than with FAS alone (Fig. 2.8.BC, three Wilcoxon tests $df = 9$, $p < 0.001$). This was because in treatments with the mixed FAS/RAS approach, using FAS for the first 500 generations resulted in the abundance of agents that could become members of the optimal team (Fig. 2.6.D). These agents could then be efficiently used to compose optimal teams using RAS in the following 500 generations. In addition, for the first 500 generations there was high variation between the evolving teams typical for FAS, which was then decreased by applying RAS (Fig. 2.9). Overall, this mixed approach overcame the convergence issues of both FAS and RAS.

We fixed the moment of switch from FAS to RAS at the middle of the evolutionary process (500th generation). But if needed one may envision a dynamic switch between the agent swapping methods, which should be performed as soon as the variation between teams stabilizes (here around the 250th generation, Fig. 2.9). In addition,

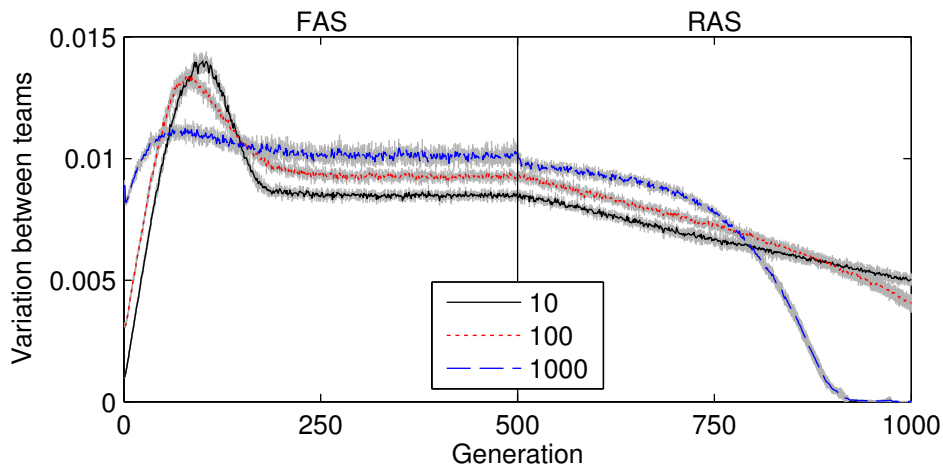


Figure 2.9: Mean variation (\pm s.d. in grey) between the 1000 teams in a population, quantified with the standard deviation of teams’ performance in a population. The teams were evolved in three different conditions (10, 100 and 1000 groups of identical agents in the optimal composition) with free agent swapping in generations from 1 to 500 (inclusive), and with restricted agent swapping in generations from 501 to 1000 (10 replicates). The black, solid vertical line at generation 500 marks the transition from using FAS to using RAS.

we tested alternative implementations of combining RAS and FAS, by applying both operators simultaneously but with different and varying probabilities of agent swapping. The results indicate that RAS and FAS should be used exclusively in order to balance the exploration and exploitation aspects of the evolutionary algorithm (*Supplementary materials*, Section 2.7.6).

2.6 Discussion

We considered a situation when an optimal team consisted of equally sized groups of identical agents, but the results allow us to speculate how the restricted and free agent swapping compare in the situation when groups differ in sizes. In treatments with RAS, team performance depended on the contents of the initial population, and not on the actual optimal composition. Consequently, limitations and analysis presented in this Chapter for RAS should also hold for optimal team composition with groups of different sizes. This is because no link should be expected between the number of agents of a specific type in the initial population and in the optimal composition.

In contrast, with FAS, team performance depended on the optimal team composition. FAS introduced a variation between team compositions, which was detrimental to

team performance. The level of this variation depended on the condition. For example, the variation was the highest, and hence team performance was the lowest, when the optimal team was composed of many small groups (here 1000 groups of 1 agent). This was because, with FAS, it is more probable for a team to deviate from the optimum when the optimal composition consists of small groups than when it consists of big groups. We illustrate this property with a simple thought experiment. Consider two teams of 4 agents, both displaying the same optimal composition that consists of four groups of 1 agent. There are 16 possible free agent swaps between the two teams. Only 4 of them, the ones that happen on the corresponding positions, maintain the optimal composition in the teams. In contrast, consider two teams of 4 agents, both displaying the same optimal composition that consists of two groups of 2 agents. Again, there are 16 possible free agent swaps between the two teams. But now there are 8 swaps that maintain the optimal composition in the teams. Therefore, the performance of teams evolved with FAS depends on the size of groups in the optimal composition. But it does not depend on the fact if these groups are of equal size, or not. Consequently, limitations and analysis presented in this Chapter for FAS should also hold for optimal team composition with groups of different sizes.

Several general guidelines on applying RAS and FAS to real-life problems can be drawn. First, we compared RAS and FAS on a problem of evolving team compositions, whose formulation was highly general. The part of a team's genotype that corresponded to a single agent was rather simplistic, i.e., a vector of one (main experiments) to a few numbers (see additional experiments in Appendix B). Consequently, our results are directly applicable whenever the goal is to optimize just few control parameters per agent, and not to create entire controllers for each agent from scratch. This is the case for many real-life applications, when the problem is often decomposed into two parts: the one solved manually and the one solved automatically. For example, in behavioral robotics, agents perform behaviors that have been implemented manually (Balch and Arkin 1998; Brooks 1991a,b). In bio-inspired scheduling, agents allocate themselves to tasks which they handle by calling existing routines (Agassounon and Martinoli 2002; Campos et al. 2000; Jones and Matarić 2003; Matarić et al. 2003). In real-time strategy games, units perform predefined actions or entire sequences of actions (Lichocki et al. 2009; Preuss et al. 2010). In ant colony optimization the agents traverse the edges of a graph according to established algorithms (Bonabeau et al. 2000; Dorigo et al. 2006). In all these cases, the agents choose behaviors, tasks, actions and edges based on the values of a few control parameters that can be optimized automatically (see, e.g., Lichocki et al. 2012; White et al. 1998). In conclusion, our problem's formulation, although general,

very well conforms to practical applications, as diverse as robotics, task allocation, video games and hyper-heuristics (i.e., searching in the space of heuristics).

Three implementation issues related to RAS and FAS should be discussed. First, we considered the agent swapping operators to work on teams having linear genotypes (in contrast to, e.g., a tree-based representation popular in genetic programming). Both RAS and FAS can be easily adapted to other teams' representations. In principle, FAS requires teams to be multi-sets of agents, i.e., sets in which elements may repeat. Whereas RAS requires teams to be sequences of agents. This is not much of a limitation, because one can always create a sequence by imposing an ordering on a multi-set. Note, however, that the ordering must remain constant during the evolution or, at least, it must change in the same way for all teams in the population. Otherwise, the agents could change their positions due to variations in the ordering. Consequently, there would be no restrictions on agents keeping their positions in teams and using RAS would make little sense.

Second, we evolved teams of constant size, and thus we used genotypes of constant length. Both RAS and FAS could be adapted to variable-length genotypes by restraining the operators to act only on common parts of the genotypes. Alternatively, one could use agent deletion and duplication which alters the team compositions similarly to FAS (*Supplementary materials*, Section 2.7.7).

Third, for the mixed FAS/RAS operator we considered also alternative implementations. We applied both operators simultaneously, but with different and varying probabilities of agent swapping. The results discourage such an approach and suggest that RAS and FAS should be used exclusively. This is due to the disruptive character of FAS, which is strongly marked even for low probabilities of agent swapping (*Supplementary materials*, Section 2.7.6).

We focused on the evolution of team compositions using team encoding, i.e., when a single genotype encodes all individuals from one team. Team encoding implicitly assumes team level of selection (i.e., global reward, see Fig. 2.1). In contrast to individual level of selection, team level of selection omits pathologies caused by competition between team members (e.g., evolution of cheaters, see Goings et al. 2004; Mitri et al. 2009; Waibel et al. 2011), which could decrease the overall team performance. Therefore, team level of selection, and consequently team encoding, are advocated when the goal is to optimize the overall team's performance (Waibel et al. 2009). Studying inter-agent interactions usually requires a game-theoretic perspective and is an interesting subject in itself (see, e.g., Shoham and Leyton-Brown 2009), but was not in the scope of this

Chapter.

Nevertheless, our results are also relevant for evolutionary algorithms using individual encoding (e.g., Bull 1998; Eiben et al. 2007; Floreano et al. 2007; Soule and Heckendorn 2008; Suzuki and Arita 2006; Waibel et al. 2009). This is because, the converging character of RAS and the disruptive character of FAS are general properties of these operators. Thus, premature and insufficient convergence would be to some extent marked with RAS and FAS, respectively, regardless of the encoding and the level of selection used. For example, with individual encoding, in order to assess performance the agents are grouped into teams, often at random (see, e.g., Floreano et al. 2007; Waibel et al. 2006, 2009). Random grouping in individual encoding introduces a constant variation between compositions of evolving teams, similarly to FAS in team encoding. Consequently, random grouping shares the limitations of FAS, and may hamper the evolutionary process whenever high level of genetic specialization between agents is required.

It should be noted that the evidence from our study has three limitations, which translate into three directions of future research. First, we assumed that a single change in team composition perfectly translates into a corresponding change in team performance. It might be desirable to consider epistatic and noisy fitness functions (see, e.g., Goldberg and Deb 1991). We expect FAS to drive the evolution towards optimum even then, because of the highly explorative nature of this method. We supported this claim by validating agent swapping operators in a stochastic and dynamic problem of decentralized task-allocation (Appendix B). In this practical application, we showed that the teams evolved with FAS had higher performance than the teams evolved with RAS, probably due to low level of agent specialization that was required to solve the problem. This result is consistent with our previous analyses and shows that the conclusions are valid also for more complex problems.

Second, we assumed a situation with no locally optimal team compositions. In practice, the globally optimal team composition might be hidden in a part of the landscape with low average payoff (i.e., deceptive fitness landscape, see De Jong 1993). Such a fitness landscape could hamper the evolution of optimal teams, similarly to other problems being solved with evolutionary algorithms (Horn and Goldberg 1994; Whitley 1991). Fortunately, evolutionary algorithms frequently turn out to be excellent heuristics for most deceptive fitness landscapes (De Jong 1993).

Third, we focused on teams consisting of agents that did not evolve themselves.

This allowed us to decouple the effects of agent recombination and agent swapping and to directly study the evolution of team compositions. We also validated RAS and FAS including mutations. This did not affect the conclusions drawn (*Supplementary Materials*, Section 2.7.3 and Appendix B). Note that developing the methods that efficiently evolve large teams of complex genotypes for multi-agent systems remains a big challenge. So far, other authors have focused on the evolution of complex controllers for small teams of agents using, e.g., genetic programming (Haynes and Sen 1996; Luke and Spector 1996) and neuroevolution (Floreano et al. 2007; Waibel et al. 2009), and have overlooked the evolution of team compositions (with the exception of Hara and Nagao 1999 and Bongard 2000a who, however, considered only simple teams of up to 6 groups). Consequently, our investigation of evolving team compositions fills the gap and hopefully will help to address other challenging questions on optimizing multiple agents.

Overall, in this Chapter, we were able to quantify for the first time the efficiency of restricted and free agent swapping in the evolution of team compositions using team encoding. The analysis and comparison between restricted and free agent swapping revealed the limitations of both approaches. Our study highlights and explains the convergence issues, which were detrimental to team performance. Using restricted agent swapping resulted in an efficient evolution of team compositions, only if the population contained enough agents that could become members of the optimal team (but regardless of the optimal composition). In contrast, using free agent swapping resulted in an efficient evolution of team compositions, only if optimal composition imposed a low requirement on genetic specialization of the agents (but regardless of the contents of initial population). In order to overcome these limitations, the results strongly suggest to use a combination of the two methods of agent swapping, which were so far only used in isolation. The mixed approach balances the exploration and exploitation aspects of the evolutionary algorithm and experimentally proves to be a superior strategy in the evolution of team compositions.

2.7 Supplementary materials

2.7.1 1-point and 2-point crossover

Miconi (2003) and Nakashima et al. (2006) used 1-point and 2-point crossover to recombine the teams' genotypes. The n -point crossover might be considered as a competitive way to implement RAS. It has been shown that n -point crossover has lower exploratory power than uniform crossover (De Jong and Spears 1992; Eshelman et al. 1989). Also,

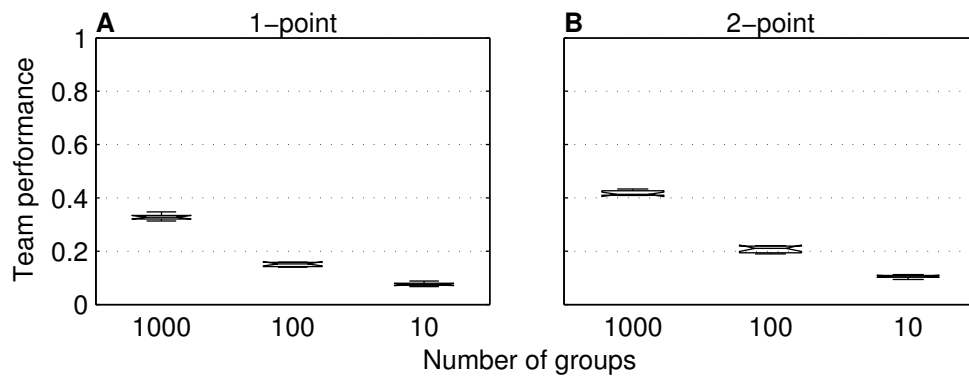


Figure 2.10: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) 1-point crossover and (B) 2-point crossover (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance.

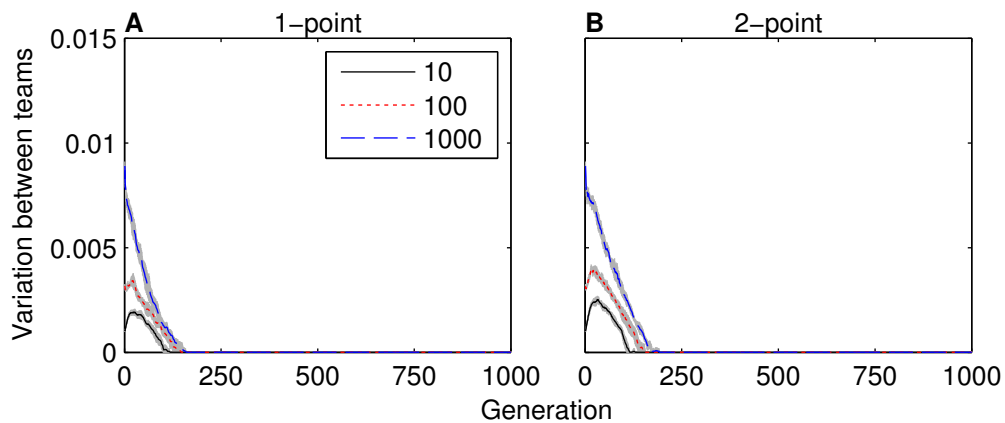


Figure 2.11: Mean variation (\pm s.d. in grey) between the 1000 teams in a population (10 replicates), measured for all 1000 generations. The variation between the teams was quantified with the standard deviation of teams' performance in a population for (A) 1-point crossover and (B) 2-point crossover, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition).

teams are sets of agents (formally multi-sets) and not sequences of agents, thus the disruptive nature of the uniform crossover (i.e., breaking the schemata, see De Jong and Spears 1992) is not necessarily an undesired feature. These two facts suggest that uniform crossover (Haynes and Sen 1996; Syswerda 1989) is a better choice than n -point crossover in the implementation of RAS. We found support for this claim, showing that in treatments with RAS implemented as 1- and 2-point crossover team performance was lower (Fig. 2.10) than in treatments with RAS implemented as uniform crossover. This performance drop was associated with faster convergence (Fig. 2.11) with 1- and 2-

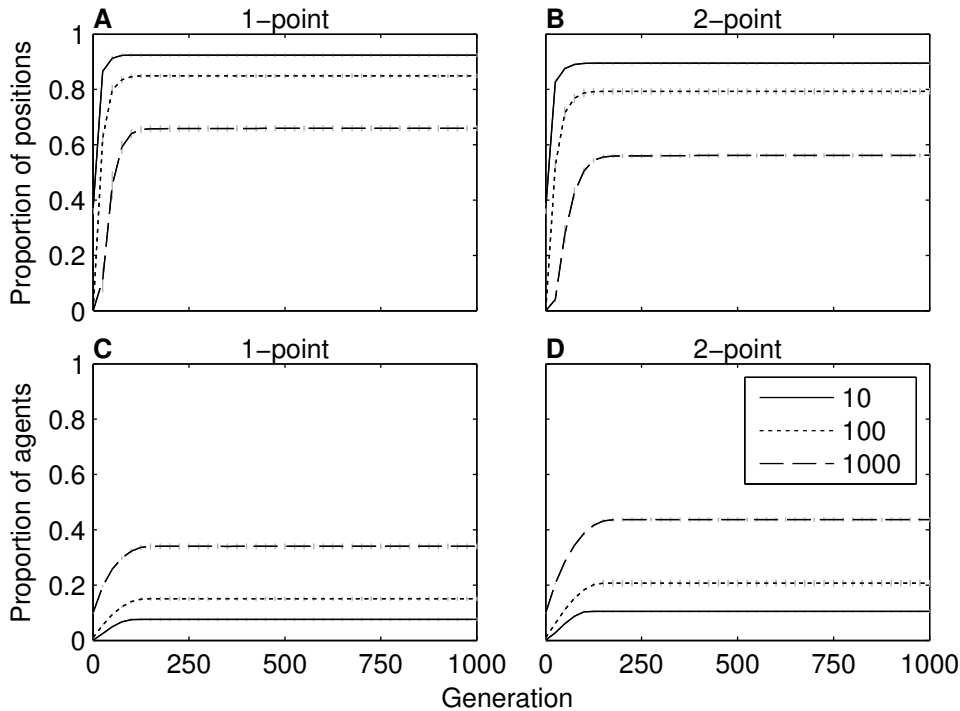


Figure 2.12: (A and B) Mean proportion (\pm s.d. in grey) of corresponding positions that contained in all 1000 teams only the agents that could not become members of the optimal team. (C and D) Mean proportion (\pm s.d. in grey) of agents in a team that could become members of the optimal team, averaged over 1000 teams. The populations were analyzed every 25 generations over 10 replicates for (A and C) 1-point crossover and (B and D) 2-point crossover, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition).

point than uniform crossover. Consequently, the proportion of corresponding positions that in all teams contained only agents that could not become members of the optimal team was higher (Fig. 2.12.AB) with 1- and 2-point than uniform crossover. And, the proportion of agents in teams that could become members of the optimal team was lower (Fig. 2.12.CD) with 1- and 2-point than uniform crossover.

In treatments with 1- and 2-point crossovers, to construct the 1000 teams of the following generation we 500 times selected two teams (tournament size set to 2). Then, instead of reorganizing the two teams with uniform crossover, the teams' genotypes were crossed-over with 1-point or 2-point crossovers. With 1-point crossover, one random locus $1 \leq L \leq 1000$ was chosen and the parenting teams exchanged all agents on positions from 1 to L . With 2-point crossover, two random loci $1 \leq L_1 \leq L_2 \leq 1000$ were chosen and the parenting teams exchanged all agents on positions from L_1 to L_2 . Other settings were the same as in the main experiments (see Section 2.4).

2.7.2 Effects of changing agent swapping probability

To test the sensitivity of the results to different probability of swapping the agents between the parenting teams we performed an additional experiment with this probability set to 0.2. Other settings were the same as in the main experiments (see Section 2.4). We found that the control experiment is in a reasonably good agreement with the main results (Section 2.5), with respect to team performance (Fig. 2.13), the variation between

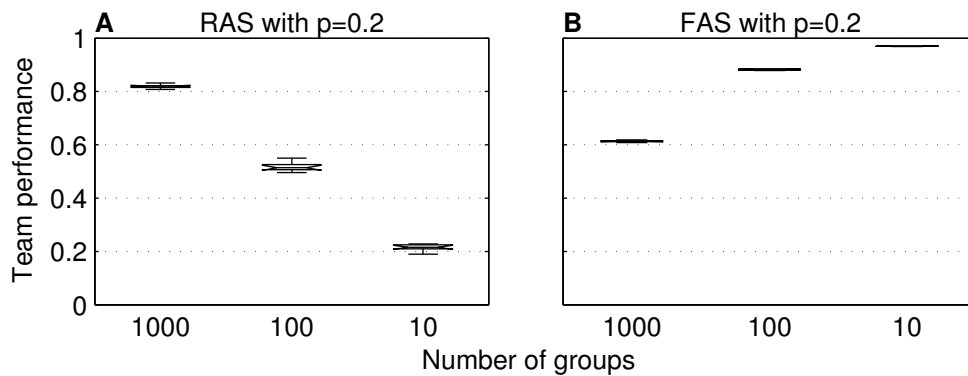


Figure 2.13: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance. For both treatments (RAS and FAS), the agents were swapped with probability 0.2.

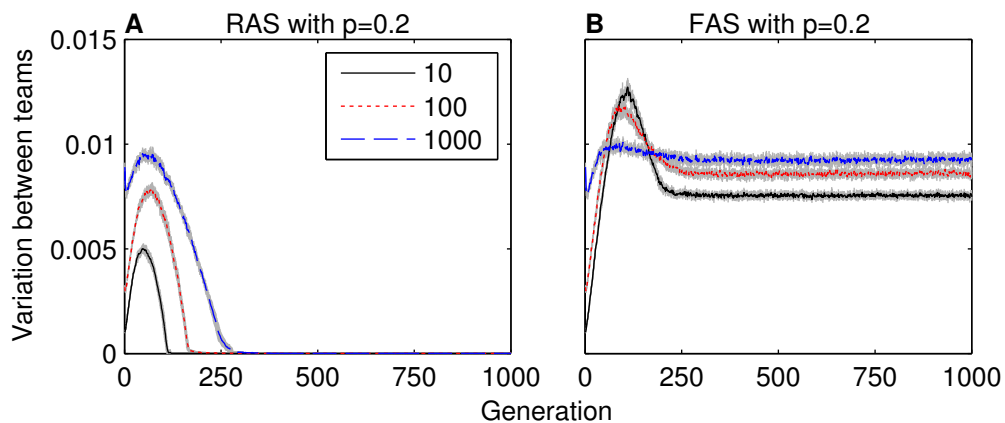


Figure 2.14: Mean variation (\pm s.d. in grey) between the 1000 teams in a population (10 replicates), measured for all 1000 generations. The variation between the teams was quantified with the standard deviation of teams' performance in a population for (A) restricted agent swapping and (B) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), the agents were swapped with probability 0.2.

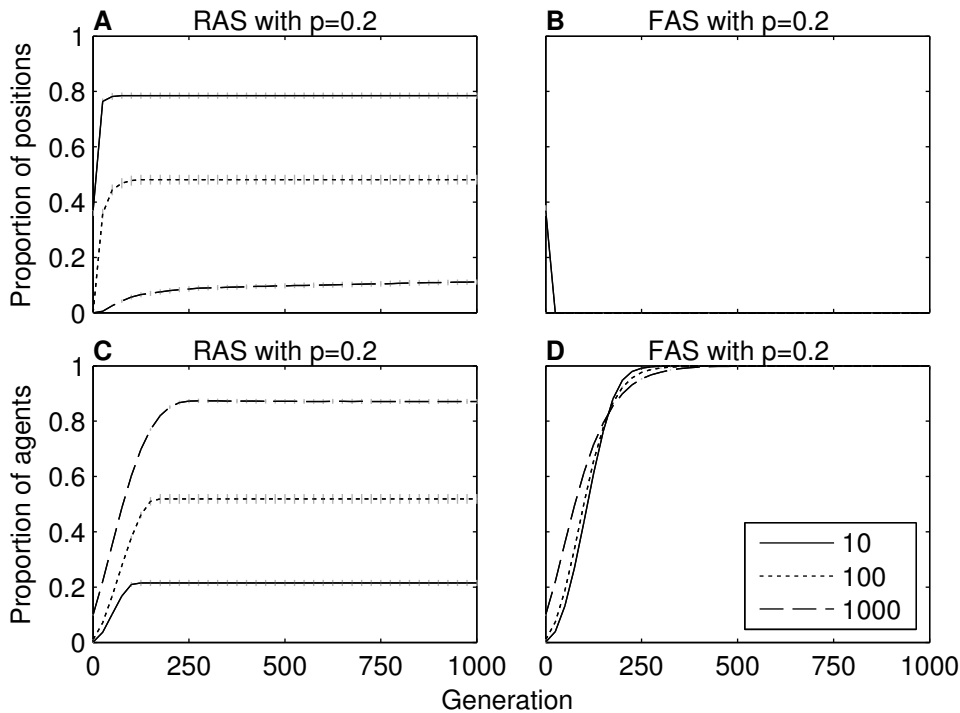


Figure 2.15: (A and B) Mean proportion (\pm s.d. in grey) of corresponding positions that contained in all 1000 teams only the agents that could not become members of the optimal team. (C and D) Mean proportion (\pm s.d. in grey) of agents in a team that could become members of the optimal team, averaged over 1000 teams. The populations were analyzed every 25 generations over 10 replicates for (A and C) restricted agent swapping and (B and D) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), the agents were swapped with probability 0.2.

teams (Fig. 2.14), the proportion of corresponding positions that in all teams contained only agents that could not become members of the optimal team (Fig. 2.15.AB) and the proportion of agents in teams that could become members of the optimal team (Fig. 2.15.CD).

2.7.3 Effects of mutation

With RAS the evolutionary algorithm suffered from premature convergence. Therefore, using mutation with RAS should have a positive effect on the exploration aspect of the evolutionary algorithm, and, thus, translate into an increase in team performance. In contrast, with FAS the evolutionary algorithm suffered from insufficient convergence. Consequently, using mutation with FAS should have a negative effect on the exploitation aspect of the evolutionary algorithm, and, thus, translate into a decrease in team

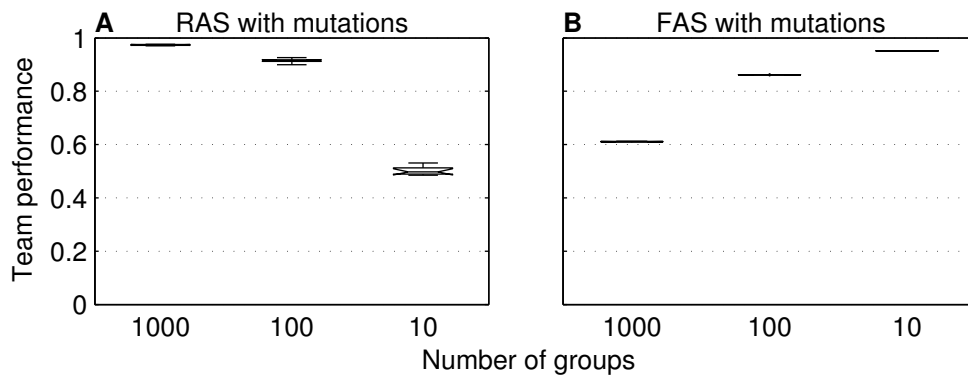


Figure 2.16: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance. For both treatments (RAS and FAS), each allele in a team's genotype was mutated with probability 0.001.

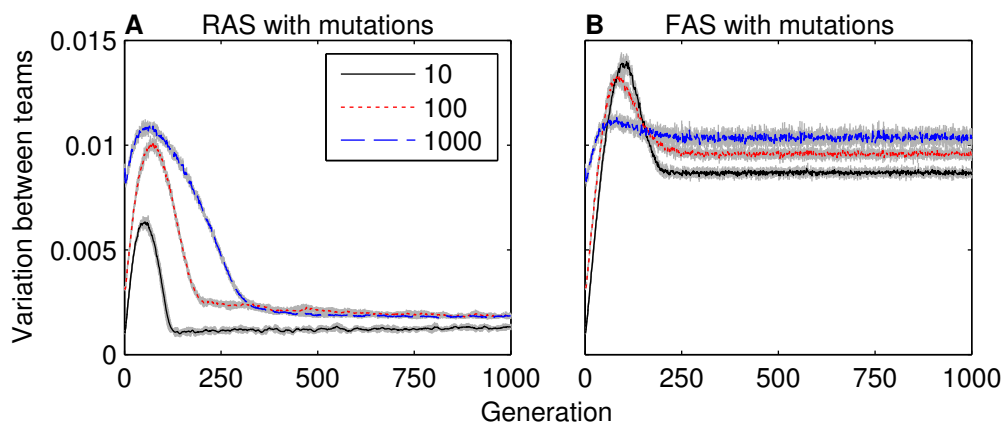


Figure 2.17: Mean variation (\pm s.d. in grey) between the 1000 teams in a population (10 replicates), measured for all 1000 generations. The variation between the teams was quantified with the standard deviation of teams' performance in a population for (A) restricted agent swapping and (B) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), each allele in a team's genotype was mutated with probability 0.001.

performance. We found support for both claims in an additional experiment where the teams' genotypes were subject to mutation, i.e., each allele of the newly added teams to the population was randomly set to a value between 1 and 10000 with a probability 0.001. Other settings were the same as in the main experiments (see Section 2.4).

In treatments with RAS, using the mutation counterbalanced to some extent pre-

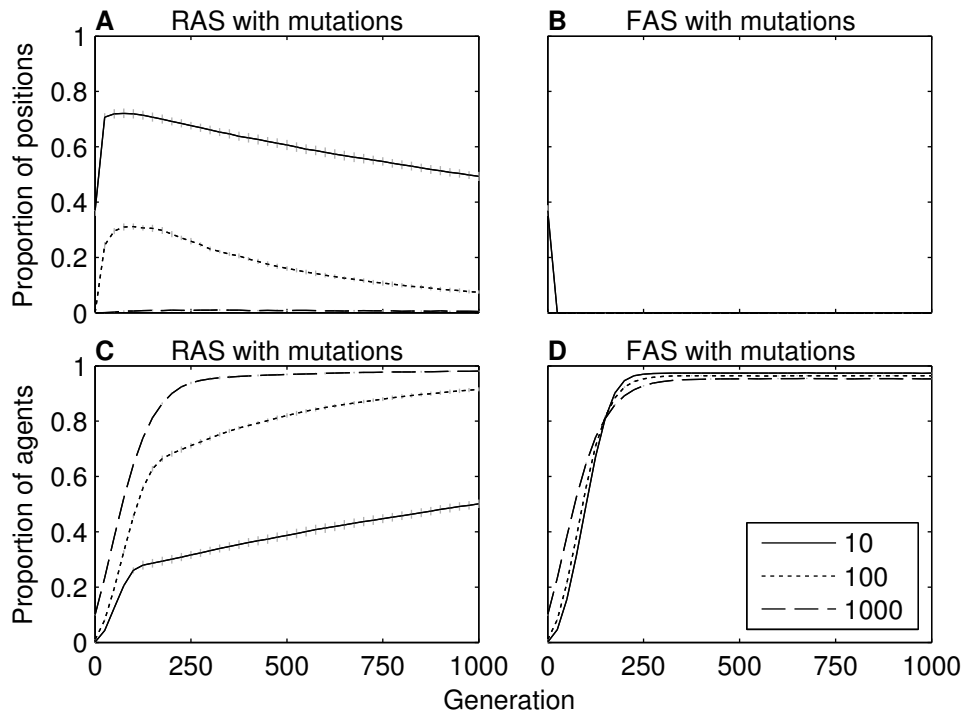


Figure 2.18: (A and B) Mean proportion (\pm s.d. in grey) of corresponding positions that contained in all 1000 teams only the agents that could not become members of the optimal team. (C and D) Mean proportion (\pm s.d. in grey) of agents in a team that could become members of the optimal team, averaged over 1000 teams. The populations were analyzed every 25 generations over 10 replicates for (A and C) restricted agent swapping and (B and D) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), each allele in a team's genotype was mutated with probability 0.001.

mature convergence (Fig. 2.17.A) and its negative effect on team compositions, with respect to the proportion of corresponding positions that contained in all teams only agents that could not become members of the optimal team (Fig. 2.18.A) and to the proportion of agents in teams that could become members of the optimal team (Fig. 2.18.C). This translated into a higher team performance (Fig. 2.16.A) than in treatments with RAS without mutation. In contrast, in treatments with FAS the mutation increased the variation between the teams (Fig. 2.17.B), which had a detrimental effect on evolving team compositions (Fig. 2.18.BD) and, consequently, on team performance (Fig. 2.16.B).

2.7.4 Effects of selection pressure

We investigated how our results were affected by increasing the selection pressure. To this aim we used a tournament size set to 7. Other settings were the same as in the main

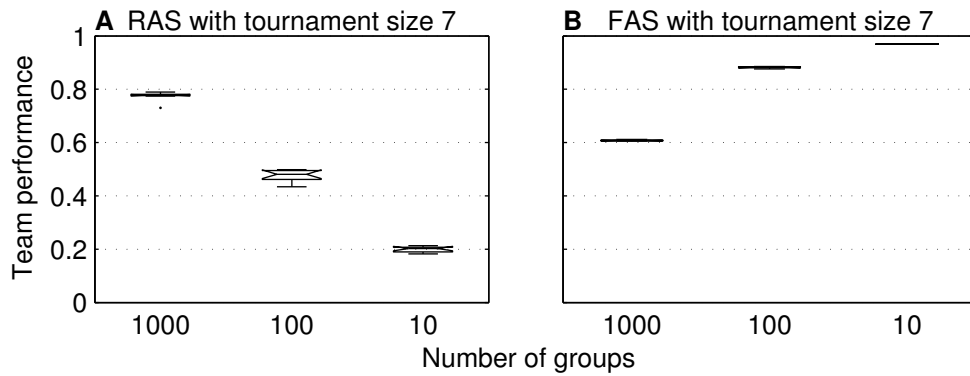


Figure 2.19: Box and Whisker plots showing the mean performance of 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance. For both treatments (RAS and FAS), the tournament size in the tournament selection was set to 7.

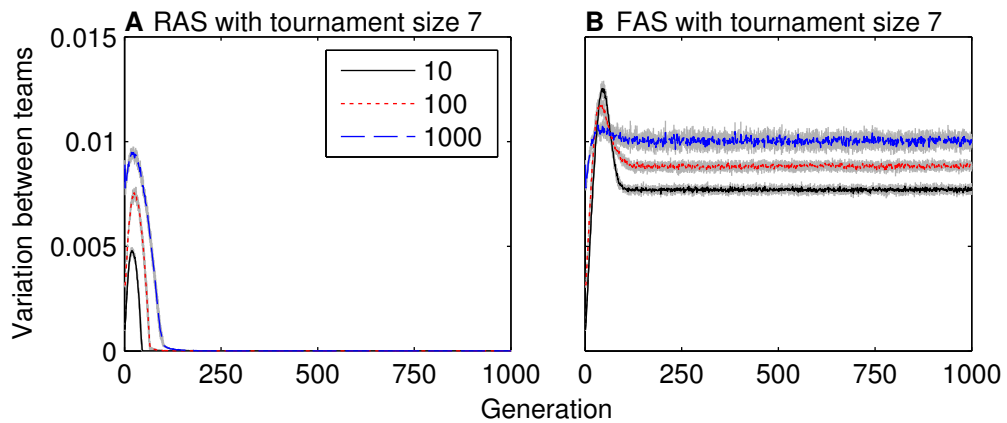


Figure 2.20: Mean variation (\pm s.d. in grey) between the 1000 teams in a population (10 replicates), measured for all 1000 generations. The variation between the teams was quantified with the standard deviation of teams' performance in a population for (A) restricted agent swapping and (B) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), the tournament size in the tournament selection was set to 7.

experiments (see Section 2.4). We found that with RAS, stronger selection increased premature convergence (Fig. 2.20.A), which translated into lower team performance (Fig. 2.19.A) than in treatments with tournament size set to 2 (compare with Fig. 2.4). This was because the increased premature convergence lead to higher proportion of corresponding positions that in all teams contained only agents that could not become members of the optimal team (Fig. 2.21.A) and to lower proportion of agents in teams that could become members of the optimal team (Fig. 2.21.C).

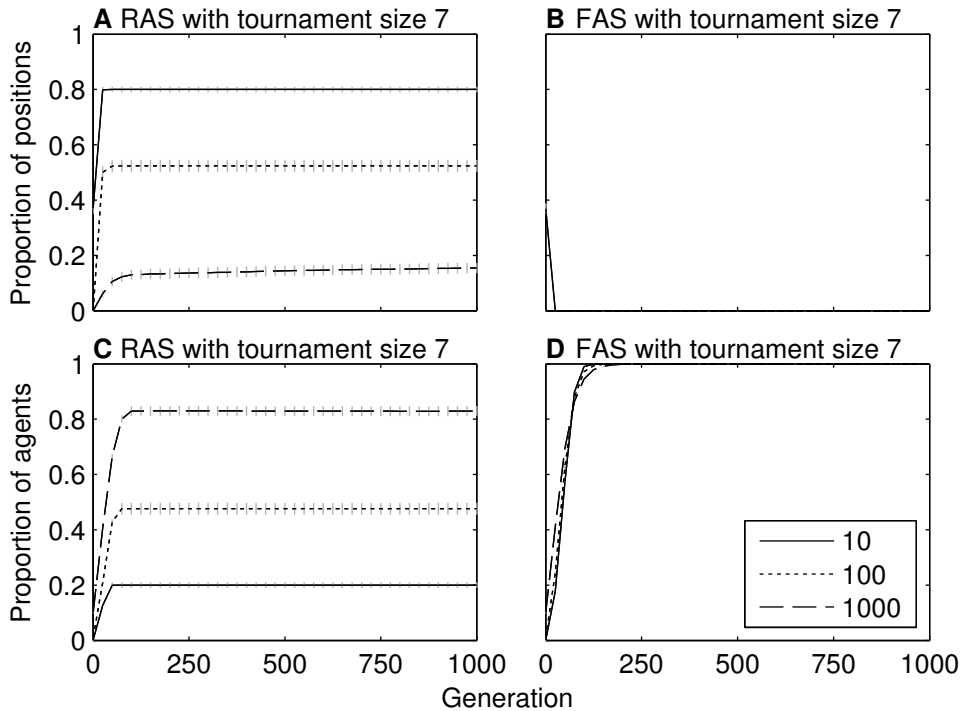


Figure 2.21: (A and B) Mean proportion (\pm s.d. in grey) of corresponding positions that contained in all 1000 teams only the agents that could not become members of the optimal team. (C and D) Mean proportion (\pm s.d. in grey) of agents in a team that could become members of the optimal team, averaged over 1000 teams. The populations were analyzed every 25 generations over 10 replicates for (A and C) restricted agent swapping and (B and D) free agent swapping, in each of the three conditions (10, 100 and 1000 groups of identical agents in the optimal composition). For both treatments (RAS and FAS), the tournament size in the tournament selection was set to 7.

In contrast, with FAS, stronger selection had no effect on the performance of the evolved teams (Fig. 2.19.B), because it did not overcome the problem of insufficient convergence (Fig. 2.20.B). Consequently, the results for treatments with FAS with the two strengths of the selection pressure are in the agreement with respect to the proportion of corresponding positions that in all teams contained agents that could not become members of the optimal team (Fig. 2.21.B) and the proportion of agents in a team that could become members of the optimal team (Fig. 2.21.D).

2.7.5 Best team performance in the population

We report here the performance of the best team at generation 1000th over 10 replicates (Fig. 2.22) and compare it with mean team performance. To this aim we calculated over 10 replicates the mean \pm s.d. ratio r between the value of mean performance

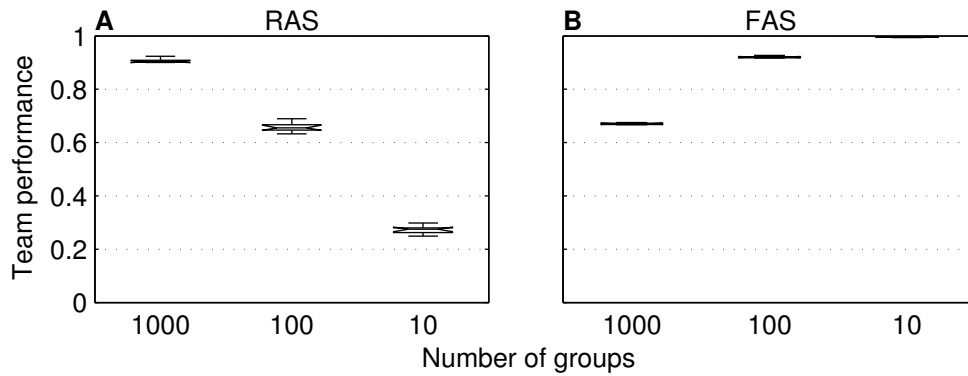


Figure 2.22: Box and Whisker plots showing the best performance from 1000 teams evolved with (A) restricted agent swapping and (B) free agent swapping (10 replicates). Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance.

in the population and the value of the best performance. There were no differences between mean and best team performance for treatments with RAS ($r = 1 \pm 0$ for all three conditions). There were small differences between mean and best team performance for treatments with FAS (r for 1000 groups: 0.931 ± 0.004 ; 100 groups: 0.955 ± 0.003 ; 10 groups: 0.971 ± 0.001). These differences between mean and best team performance correspond well to the variation between teams at the 1000th generation (0 for RAS, and approx. 0.01 for FAS, see Fig. 2.5).

2.7.6 Alternative methods of combining team crossover operators

In the main experiment we combined FAS with RAS by applying them from generation 1 to 500 and from generation 501 to 1000, respectively (Section 2.5.3). Here, we investigated two alternative methods that apply both operators together, but with different probabilities of agent swapping (Fig. 2.23.AB). With the first method, the probability of agent swapping with FAS (p_{FAS}) was set to = 0.5 at generation 1 and decreased with a constant rate to 0 at generation 1000. With the second method, the p_{FAS} was set to 0.5 from generation 1 to 250, then it decreased with a constant rate to 0 until the generation 750, and was fixed to 0 until the generation 1000. In both methods, the probability of agent swapping with RAS (p_{RAS}) was set to $0.5 - p_{FAS}$. Other settings were the same as in the main experiments (see Section 2.4).

The results for both alternative methods were discouraging (Fig. 2.23.CD), because, the disruptive character of FAS was strongly marked even for low probabilities of agent

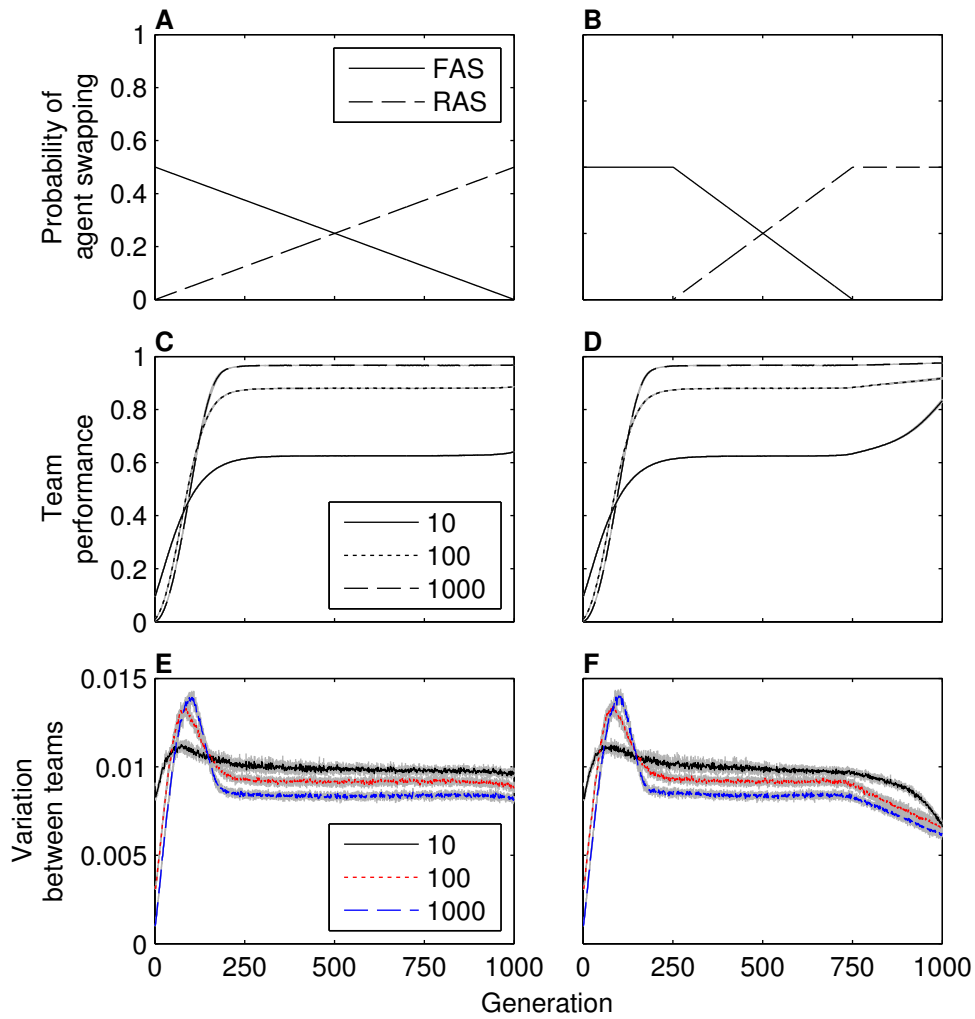


Figure 2.23: (A and B) The probabilities of agent swapping with two alternative methods of combining (A) RAS and (B) FAS. (C and D) Mean performance (\pm s.d. in grey) of 1000 teams. Teams consisted of 1000 agents, which needed to display three different compositions (10, 100 and 1000 groups of identical agents) in order for the team to achieve the optimal performance. (E and F) Mean variation (\pm s.d. in grey) between the 1000 teams. The variation between the teams was quantified with the standard deviation of teams' performance in a population. Both the mean performance and the mean variation were measured for all 1000 generations (10 replicates).

swapping. Consequently, the first alternative method did not converge (Fig. 2.23.E), and the second alternative method started converging only after p_{FAS} decreased to 0 (Fig. 2.23.F). Therefore, in order to balance well the exploration and exploitation aspects in the evolution of team compositions one should use FAS and RAS exclusively.

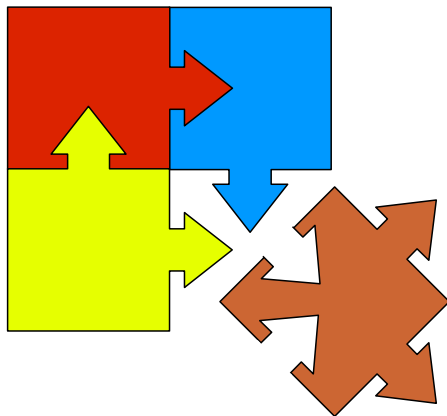
2.7.7 Other operators altering team composition

Interesting connections between RAS, FAS and other team composition altering operators exist. First note that FAS swaps agents on permuted positions, in contrast to RAS which swaps agents on corresponding positions. Consequently, one can implement FAS as a composition of RAS and an operator that permutes the agents' order in a team, i.e., samples agents without repetitions.

A natural next step is to consider a situation when the agents in a team are sampled with repetitions, because this allows for a more straightforward implementation of the agents' sampling operator. Let a parenting team have M agents. In order to construct an offspring team, the agent sampling with repetition (ASWR) picks uniformly at random M agents with repetitions from the parenting team and puts them in the offspring team. Consider agent A in the parenting team. The number of its copies in the offspring team follows a binomial distribution with M trials and the probability of success being $\frac{1}{M}$. Thus, if the size of a team is sufficiently large ($M > 20$) one can approximate the binomial distribution by Poisson distribution with the coefficient $\lambda = 1$. Consequently, after applying ASWR on a parenting team, each agent will be present in the offspring team k times with probability $\frac{1}{e \cdot k!}$, where $k = 0, 1, 2, \dots, M$. Thus, ASWR implements a sort of simultaneous agents' deletion and duplication (or rather "multiplication").

Interestingly, operators that delete and duplicate parts of genomes have been already proposed. In genetic programming there is branch deletion and duplication (Koza 1995). And, in gene expression programming there is gene transfer (Ferreira 2001). Gene transfer is equivalent to branch "substitution", i.e., having two branches A and B , delete A and duplicate B in place of A . In the context of evolving team compositions an analogous agent transfer operator might be proposed. It should have a similar impact on the evolutionary dynamics as FAS. This is because agent transfer would be in essence similar to agent sampling, which in turn is a subcomponent of FAS. Note however, that agent duplication, deletion and transfer are applied to a single parenting team. Consequently, they should be considered as mutations, rather than cross-over.

3 Neural networks as mechanisms to regulate division of labor



In the previous chapter, we studied the evolution of functional specialization fixed at the genetic level. Here, we take a step further and focus on the flexibility of division of labor at the behavioral level. We take a biological perspective, and we quantify and compare the efficiency in task allocation between the models proposed to explain division of labor in social insects. We expose the limitations of existing models which constrain behavioral flexibility of workers and, thus, lead to suboptimal colony performance when quick and precise reallocation of workforce is required. To overcome this issue, we propose a novel task allocation algorithm that builds on top of the existing biological models. From a methodological standpoint, we provide a comprehensive framework for modeling task allocation by using artificial neural networks, thus bridging the gap between biology and engineering.

3.1 Disclosure

Contents of this chapter are taken from an article of the same title, which have been published in *The American Naturalist* (Lichocki et al. 2012). The article is authored by Paweł Lichocki, Danesh Tarapore, Laurent Keller and Dario Floreano. Paweł Lichocki developed formal analyses, implemented and executed simulations. All authors designed the project, discussed the results and wrote the manuscript. Ana Duarte, Sara Mitri, Robert Page, Elva Robinson, Gene Robinson, Markus Waibel, Steffen Wischmann, two anonymous reviewers and two editors contributed useful comments on the manuscript. Pavan Ramdya proof checked the manuscript. The work was supported by the Swiss National Science Foundation and an ERC advanced grant.

3.2 Abstract

In social insects, workers perform a multitude of tasks such as foraging, nest construction and brood rearing without central control of how work is allocated among individuals. It has been suggested that workers choose a task by responding to stimuli gathered from the environment. Response threshold models assume that individuals in a colony vary in the stimulus intensity (response threshold) at which they begin to perform the corresponding task. Here, we highlight the limitations of these models with respect to colony performance in task allocation. First, we show with analysis and quantitative simulations that the deterministic response threshold model constrains the workers' behavioral flexibility under some stimulus conditions. Next, we show that the probabilistic response threshold model fails to explain a precise colony response to varying stimuli. Both of these limitations are detrimental to colony performance when dynamic and precise task allocation is needed. To address these problems we propose extensions of the response threshold models by adding variables that weigh stimuli. We test the extended response threshold model in a foraging scenario and show in simulations that it results in an efficient task allocation. Finally, we show that response threshold models can be formulated as artificial neural networks, which consequently provide a comprehensive framework for modeling task allocation in social insects.

3.3 Introduction

High levels of cooperation are often cited as the primary reasons for the ecological success of social insects (Hölldobler and Wilson 1990; Oster and Wilson 1978; Robinson 1992). In social insects, workers perform a multitude of tasks such as foraging, nest construction and brood rearing without central control of how work is allocated among individuals (Gordon 1996; Pratt 2005; Seeley 1989; Wilson and Hölldobler 1988). Experimental evidence indicates that individuals in a colony vary in their propensity to perform different tasks (Jones et al. 2004; Julian and Cahan 1999; Kryger et al. 2000; Oldroyd et al. 1993; Robinson et al. 2009). It has been suggested that workers choose a task by responding to stimuli gathered from the environment (e.g., presence or absence of food) or from interactions with other workers (Bonabeau et al. 1996).

Empirical studies have identified a variety of factors affecting the stimulus intensity (response threshold) at which workers initiate to perform a given task (Duarte et al. 2012). First, in many species there is a strong division of labor based on the age of workers, a phenomenon designated age polytheism (Oster and Wilson 1978; Wilson 1971). Second, size and morphology is also strongly correlated with the likelihood of workers to undertake the various tasks in species with distinct morphological castes (Wilson 1980). Third, in species with multiply-mated queens or multiple queens per colony, different patriline and matriline tend to differ in their tendencies to perform certain tasks demonstrating a genetic component in response threshold (e.g., Robinson and Page 1988). Finally, individual experience has been shown to influence task preference in the ant *Cerapachys biroi* (Ravary et al. 2007).

The combined effects of age, size, genetic background and individual experience should lead to substantial intra-colony variation in response thresholds. The resulting individual difference in the response of workers to a given stimulus intensity leads to individuals with a lower threshold for a given task being more likely to perform that task than individuals with a higher threshold. A variety of models have been proposed to account for the emergence of intra-colony division of labor on the basis of variation in response thresholds (Bonabeau et al. 1996; Graham et al. 2006; Jeanson et al. 2007; Page Jr and Mitchell 1998; Robinson 1987a, 1992; Theraulaz et al. 1998; see Beshers and Fewell 2001; Duarte et al. 2011; Smith et al. 2008b for reviews).

The two most often used models are the deterministic response threshold model (DTM) (Page Jr and Mitchell 1998), and the probabilistic response threshold model (PTM) (Bonabeau et al. 1996). Both models assume that all workers receive information

of the colony needs via commonly perceived stimuli. With the DTM each worker performs the task with the highest positive difference between the stimulus and its own corresponding response threshold. If all the stimuli are lower than the corresponding thresholds the worker remains idle. With the PTM the relation between stimulus and threshold is interpreted as a probability to perform the task. While these response threshold models are frequently used to explain division of labor in colonies of social insects (Bertram et al. 2003; Graham et al. 2006; Jeanson et al. 2007), no attempts have been made to quantify their efficiency in task allocation. Here we show with analysis and quantitative simulations that DTM (Page Jr and Mitchell 1998) and PTM (Bonabeau et al. 1996) lead to sub-optimal colony performance under some stimulus conditions. To overcome these problems we propose an extended response threshold model (ETM) that can result in an efficient task allocation for any stimulus conditions. We experimentally compare all models by means of directed evolution (Floreano and Keller 2010; Nolfi and Floreano 2000) in a foraging scenario that requires a dynamic re-allocation of workers to different tasks according to colony needs (Tarapore et al. 2010). Finally, we show that the response threshold models can be formulated as artificial neural networks (Haykin 1998; McClelland et al. 1986), which consequently constitute a comprehensive framework for modeling task allocation in social insects.

3.4 Methods

3.4.1 Task allocation mechanism

We assumed that workers receive information of the colony needs via commonly perceived stimuli and that workers' thresholds do not vary during their lifetime. We considered a colony composed of 1000 workers facing two distinct tasks. In DTM every worker has two thresholds corresponding to each of the two tasks. A worker performs the task with the highest positive difference between the stimulus and its own corresponding response threshold, or remains idle if both of its thresholds are higher than the stimuli (Jeanson et al. 2007; Page Jr and Mitchell 1998). If the difference between the stimulus and the worker's corresponding response thresholds is the same for all tasks, one of them is randomly chosen and performed by the worker. In PTM, every worker has two thresholds corresponding to each of the two tasks and the difference between stimulus and corresponding response threshold constitutes the probability of being engaged in the corresponding task. Our probabilistic model is very similar to the original probabilistic response threshold model (Bonabeau et al. 1996), as both models lead to similar colony performance and patterns of division of labor (*Supplementary*

materials, Section 3.7.3). In ETM, every worker has two thresholds corresponding to each of the two tasks and two weights corresponding to each of the two stimuli. A worker performs the task with the highest positive difference between the weighted stimulus and its own corresponding response threshold, or remains idle if both of its thresholds are higher than the weighted stimuli. If the difference between the weighted stimulus and the worker's corresponding response threshold is the same for all tasks, one of them is randomly chosen and performed by the worker. For formal definitions of all models see (Supplementary materials, Section 3.7.1).

The response threshold models (DTM and PTM) and the extended response threshold model (ETM) can all be formulated as a more general class of parallel distributed processing models, known as artificial neural networks (Haykin 1998; Lek and Guégan 1999; McClelland et al. 1986). Artificial neural networks have been successfully used to control the behaviour of individuals in a colony (see, e.g., Floreano et al. 2007; Waibel et al. 2009) making it a useful approach to consider in modeling task allocation in social insects. An artificial neural network is a computational model that consists of a set of

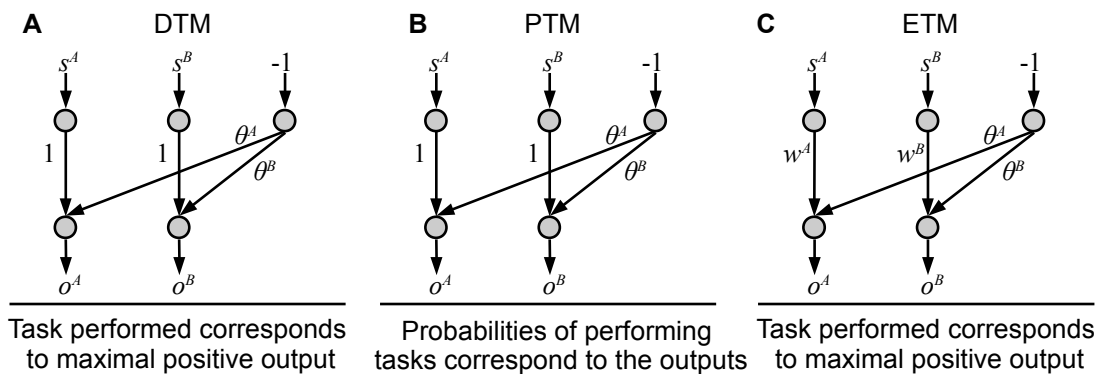


Figure 3.1: Task allocation mechanisms: A) deterministic response threshold model (DTM), B) probabilistic response threshold model (PTM), C) extended response threshold model (ETM). Consider a colony composed of multiple workers facing two tasks: A and B. s^A stands for the task A stimulus, s^B for the task B stimulus, θ^A and θ^B are the corresponding thresholds. Outputs are the sum of the weighted inputs and are calculated as $o^A = 1 \cdot s^A - \theta^A$ and $o^B = 1 \cdot s^B - \theta^B$ (for DTM and PTM), and as $o^A = w^A \cdot s^A - \theta^A$ and $o^B = w^B \cdot s^B - \theta^B$ (for ETM). Note that ETM uses two additional variables, namely w^A and w^B which are the weights for the connections between corresponding input and output neurons. In the case of the DTM and PTM these weights are constant and set to +1. In A and C each worker collects items of the type corresponding to the output unit with highest positive value, or remains idle if both outputs are lower or equal to 0. In B the values of the output units are interpreted as probabilities for the worker to choose the corresponding task. For the sake of simplicity we considered a situation with only two tasks, but the presented formalism scales to any number of tasks.

units (neurons) connected by weighted links, where the response of the output units is the sum of weighted inputs (McClelland et al. 1986). In DTM and PTM, each stimulus constitutes an input, the thresholds are the weights of the additional input constantly set to -1 and the allocated task for the worker is derived from the output neurons (Fig. 3.1AB). While in DTM and PTM the weights for task stimuli are set to $+1$, in ETM they can vary between workers (Fig. 3.1C). In DTM and ETM a worker performs the task corresponding to the output unit with highest positive value, or remains idle if both outputs are lower or equal to 0. In PTM the values of the output units are interpreted as probabilities of performing the corresponding tasks.

In artificial neural networks with so-called hidden neurons, a non-linear activation function that transforms the output of the neuron is often used. Because the artificial neural networks used here do not need non-linear activation functions and do not have hidden units, we do not mention such a function explicitly. Mathematically speaking, we consider the activation function to be the identity function. If the neuronal formalism gets extended, one may use the activation function, depending on the needs of the architecture.

3.4.2 Colony tasks

To quantify the workers' efficiency in task allocation we used a stochastic agent-based simulation to model a situation in which workers had to perform two distinct tasks (Tarapore et al. 2010). Our aim was to mimic situations with two vital tasks such as foraging and regulation of nest temperature. If the colony is efficient in foraging but does not regulate nest temperature well, the brood may die. Conversely, if nest temperature is well regulated, but little food is collected, only few offspring can be reared. A colony consisted of 1000 workers placed in an environment with an infinite number of two types of items, foraging and regulatory. The colony lifespan was divided into 100 time-steps. At the beginning of each time-step, a worker was presented with two task stimuli, one for the foraging items and the other for the regulatory items. If there were no items in the nest, the corresponding stimulus was set to its maximal intensity, which was 1. Otherwise, the intensity of the stimulus for each task was inversely proportional to the number of corresponding items in the nest. At each time step, every worker performed the chosen task (or stayed idle) according to the task allocation mechanism (DTM, PTM, ETM) considered in the experiment. At each time step, a worker had a probability of 0.1 to successfully collect one item corresponding to the task performed and, at each time-step, the number of foraged and regulated items in the nest were depleted by

ten items with an expected probability of 0.4. For formal definitions of the stimulus dynamics and the foraging scenario see (Appendix C).

The colony performance directly depended on the number of collected foraging items, but these were counted only when the number of regulatory items in the nest was within predefined bounds (140 – 160 items). At the first time-step of a simulation, there were no items of the foraging and regulatory tasks in the nest. The colony performance f was calculated by adding the partial performance obtained at each time-step, with $f = \sum_{t=1}^{100} f(t)$ where the colony performance at each time-step ($f(t)$) was quantified as the number of items of the foraging task collected when the number of items of the regulatory task present in the nest was between 140 and 160: $f(t) = b(t) \cdot g^F(t)$ where $b(t) = 1$ if $140 \leq g^R(t) \leq 160$ and $b(t) = 0$ otherwise, $g^F(t)$ represents the number of items foraged at time-step t and $g^R(t)$ the number of items being regulated within the nest at time-step t . Thus, if colonies performed well in only one of the two tasks, their performance was low. We normalized the resulting performance values by 10000, which is the expected amount of foraging items collected if all 1000 workers were foraging for all 100 time-steps with the probability of success equal to 0.1.

In all simulations the regulatory bounds were constantly fixed to the same values (140 – 160 items) (Tarapore et al. 2010). Changing the regulatory bounds would not qualitatively affect our results. First, the regulatory bounds have to be narrow for the regulatory task to require a dynamic reallocation of workers. Thus, the difference between the regulatory upper bound and the regulatory lower bound should not be varied. Second, placing the regulatory bounds on a different level (e.g., 100 – 120 or 200 – 220 items) would change only the relative length of the “initialization” phase (i.e., when the regulatory items are accumulated to reach the lower bound for the first time) and the “control” phase (i.e., when the regulatory items are maintained within the regulatory bounds). Such a change does not qualitatively affect any of the models’ properties that we investigated.

The definition of the colony performance used here lead to a strong influence of the efficient performance of the regulatory task on the benefits of foraging. It is likely that the efficient performance of two tasks frequently does not interact so strongly under natural conditions. However, we used such a strong on-off transition to get clearer results on the processes regulating the efficient performance of two complementary tasks. In some cases such a strong on-off transition may also occur in natural situations, for example as a result of dramatic variation in temperature affecting brood survival or colony response to flooding (which requires rapid movement of all individuals to avoid

colony extinction). Note also that, for simplicity we assume no conflict of interest among colony members about task allocation (i.e., the performance of all colony members is proportional to overall colony performance). This would, for example, be the case of a species where workers are completely sterile or when task performance does not affect the likelihood of an individual to reproduce in the future.

3.4.3 Genetic architecture, selection and reproduction

In experiments with DTM each worker had a genome consisting of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible real values with a resolution of $\frac{1}{128}$). In experiments with PTM each worker had a genome consisting of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible real values with a resolution of $\frac{1}{128}$). In experiments with ETM each worker had a genome consisting of two thresholds and two weights, all ranging from -1 to $+1$ (8-bit encoding, 256 possible real values with a resolution of $\frac{1}{128}$).

Although threshold values are typically expected to be non-negative (Bonabeau et al. 1996; Graham et al. 2006; Jeanson et al. 2007; Page Jr and Mitchell 1998) we allowed values between -1 to $+1$ in order to be consistent with the formalism of neural networks weights. We conducted a control experiment for DTM with thresholds in the range from 0 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{256}$) to make sure that our choice did not significantly affect the result of the experiments. We support this claim in (*Supplementary materials*, Section 3.7.3), showing that the different range of threshold values does not importantly alter the division of labor patterns and colony behavior in the treatments with DTM.

Artificial selection was conducted in thirty independent replicates for each of the three treatments. We used populations of 1000 colonies, each consisting of 1000 workers. At the first generation of each replicate, the alleles of all 1 000 000 workers were set randomly to one of the 256 values between -1 and $+1$ with a resolution of $\frac{1}{128}$. To construct the 1000 colonies of the following generation we selected the 300 colonies with the highest performance. This selected pool of 300×1000 workers was used to create the next generation of workers by using selection proportional to performance (i.e., the probability of a worker to contribute to the next generation was proportional to its colony performance). The newly created 1 000 000 workers were then randomly distributed among the new 1000 colonies of the following generation. Finally, the alleles of the workers were mutated, i.e., with a probability of 0.001 each allele was set randomly to one of the 256 values between -1 and $+1$ with a resolution of $\frac{1}{128}$.

3.4.4 Statistical analysis

To compare the task allocation efficiency for DTM, PTM and ETM we averaged, for each treatment and replicate, the performance of the 1000 colonies (30 replicates per treatment) at generation 1000. The same procedure was applied to compare other characteristics of the models. In order to understand the difference in performance between the models, we compared the proportion of workers in each of the three states (foraging, regulating, idle) averaged over all 100 time-steps (Fig. 3.3). We also compared the proportion of workers in each of the three states as a function of the time-step (Fig. 3.4). Next, we compared the number of workers' transitions between states summed over all 1000 workers and all 100 time-steps, normalized by the number of all state transitions (separately for each model). Next, we compared the time that the regulatory items were out of the predefined bounds during the entire colony lifespan (Fig. 3.6.A). We also compared the time until the regulatory items were within the predefined bounds (140-160) for the first time (Fig. 3.6.B). Finally, we compared the proportion of colonies that kept the regulatory items within the predefined bounds as a function of the time-step (Fig. 3.7). In a statistical test we compared the proportion of colonies that kept the regulatory items within the predefined bounds after they were within the predefined bounds for the first time (in order to provide a fair comparison it was performed at the 50th time-step for all treatments). Statistical significance within all treatments was determined with the Kruskal-Wallis test (nonparametric one-way analysis of variance) and between a pair of treatments with the Wilcoxon test (rank sum test for equal medians). The statistical tests were performed after the 1000th generation.

3.5 Results

3.5.1 Formal analyses

The common understanding of the response threshold models is that the workers' tendency to perform various tasks depends on its thresholds and that, by changing the threshold values, the worker can express any behavior, from generalist (switching between tasks) to specialist (dedicated to a specific task) (Beshers and Fewell 2001; Bonabeau et al. 1996; Robinson 1992). However, a mathematical analysis of the DTM reveals that the worker's behavioral flexibility depends not only on the worker's thresholds, but also on the difference between stimulus intensities. In particular, a worker can switch from task *A* to task *B*, only if there is a decrease in the difference between stimulus intensities of task *A* and task *B*. A worker can switch back from task *B* to task *A*, only if

there is an increase of the aforementioned difference (*Supplementary materials*, Section 3.7.2 for more details). Thus, contrary to the intuition standing behind the response threshold models (Robinson 1992), the workers' behaviors were influenced not only by the absolute intensities of the stimuli, but also by their relative intensities. Consequently, the values of the stimuli constrain the worker's ability to switch tasks regardless of the values of the individual thresholds. In the PTM this constraint is less marked, because the workers' responses are stochastic, thus allowing them to switch tasks more easily. However, stochastic individual responses make the response at the colony level more unreliable, even under fixed stimuli conditions (i.e., for the same stimuli intensities the response of a worker may be different, due to its random component). Thus, both the DTM and the PTM have limitations, which could be detrimental to colony performance. These problems can be overcome by extending the DTM with additional variables that weigh stimuli (ETM, Fig. 3.1). The weights relax the constraints on the flexibility of task allocation by allowing the workers to scale the stimuli if needed (see *Supplementary materials*, Section 3.7.2 for more details). At the same time, the deterministic decision rules employed in the ETM allow the workers to precisely respond to changing colony needs.

3.5.2 Simulations

To test whether the ETM allows a higher behavioral flexibility of workers and/or more precise responses at the colony level to varying stimuli, hence translating in a higher colony performance, we conducted experimental evolution with a stochastic agent-based simulation to model a situation in which workers had to perform two distinct tasks. The first was a regulatory task where workers had to maintain the amount of a given food item in their colony within predefined bounds. This would, for example, be the case of a honeybee colony maintaining about one kilogram of pollen in the hive or workers regulating within-hive temperature (Seeley 1995). The other was a foraging task where workers had to collect the highest possible amount of a second type of food item. Consequently, at a given point in time a worker could be engaged in the regulatory task, could be engaged in the foraging task (foraging worker) or could be inactive (idle worker). The performance of colonies was a function of workers being able to perform both tasks efficiently (Tarapore et al. 2010).

For each of the three models, the performance of the colonies rapidly increased within the first 200 generations of selection, and converged within 1000 generations (Fig. 3.2). However, there were important efficiency differences between the models

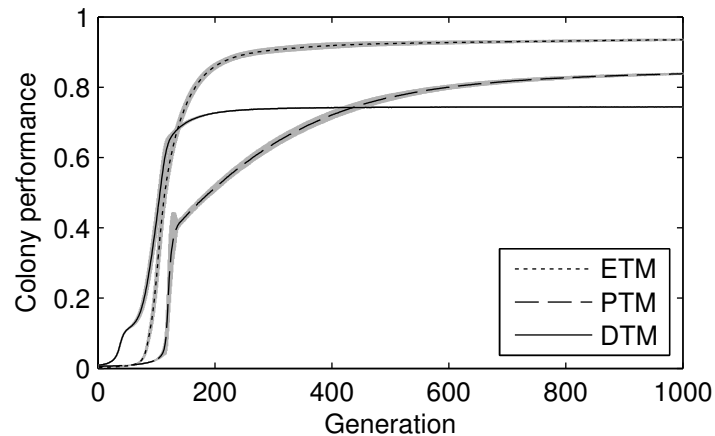


Figure 3.2: Mean \pm s.d. (in grey) performance of each of the three response threshold models over 1000 colonies (30 replicates). The performance directly depended on the number of the collected items of the foraging task, but these were counted only when the number of items of the regulatory task was within predefined bounds. Thus, the performance was high only if the workers efficiently performed both the regulatory and foraging tasks.

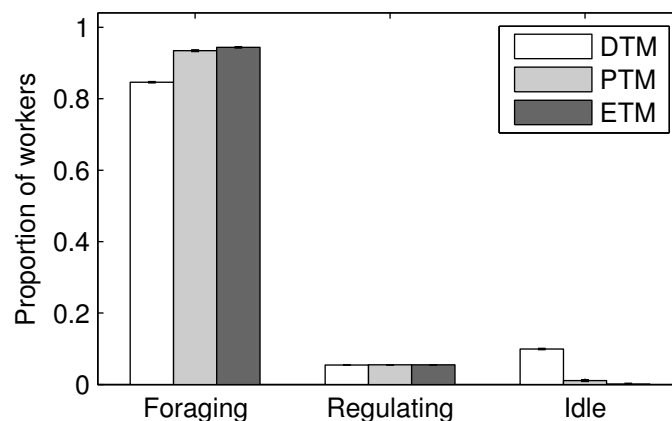


Figure 3.3: Mean \pm s.d. proportion of workers engaged in foraging, regulating or staying idle. Results are given for each of the three models, over 100 time-steps and 1000 colonies of the 1000th generation (30 replicates).

(Kruskal-Wallis test, $df = 2$, $p < 0.001$). The highest performance was achieved for the ETM (PTM: -10.3% ; DTM: -20.4% ; all pairwise Wilcoxon tests $df = 29$, $p < 0.001$).

The difference in performance between threshold models was associated with differences in the proportion of workers engaged in the foraging task (Kruskal-Wallis test, $df = 2$, $p < 0.001$). The proportion of foraging workers was the highest for the ETM (PTM: -1.1% ; DTM: -10.5% ; all pairwise Wilcoxon tests $df = 29$, $p < 0.001$; Fig. 3.3). The

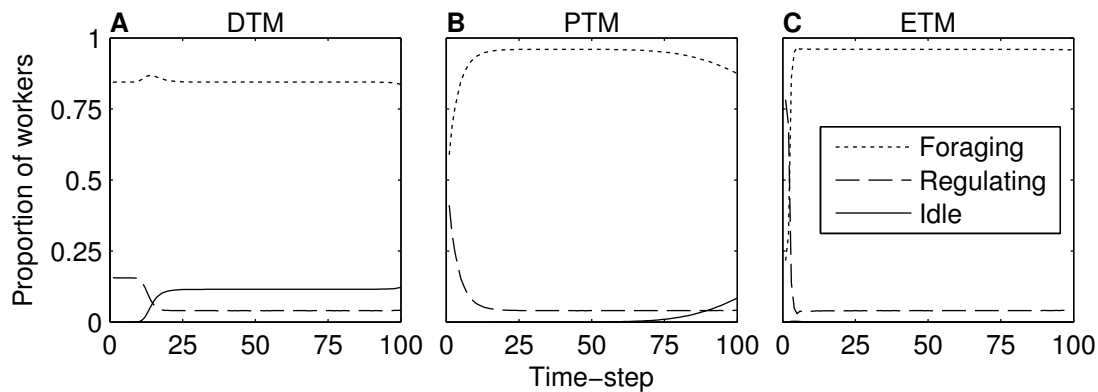


Figure 3.4: Mean proportion of workers engaged in the foraging task (short-dashed line), in the regulating task (long-dashed line) or staying idle (solid line) as a function of time-steps. Results are given for each of the three models, over 1000 colonies of the 1000th generation (30 replicates).

proportion of workers staying idle also differed (Kruskal- Wallis test, $df = 2$, $p < 0.001$) and was the lowest for the ETM (PTM: +2293.2%; DTM: +21907.5%; all pairwise Wilcoxon tests $df = 29$, $p < 0.001$; Fig. 3.3). By contrast, there was only little variation in the proportion of workers engaged in the regulatory task (Fig. 3.3), although the differences among models were significant (Kruskal-Wallis test: $df = 2$, $p < 0.001$). Furthermore, with the PTM the proportion of foraging workers was not constant during the entire colony lifespan and decreased in time, which was not observed for other models (Fig. 3.4).

The differences in workers' distribution among the three threshold models was associated with the differences in the frequency of worker transitions between being engaged in the foraging and regulatory tasks and staying idle (six Kruskal-Wallis tests, one for each type of transition: all $df = 2$, all $p < 0.001$; Fig. 3.5). In line with theoretical predictions the behavioral flexibility between the foraging and regulatory tasks was constrained with the DTM, with the effect that the workers switched much more often between the regulatory task and staying idle, than with other models (all pairwise Wilcoxon tests: $df = 29$, $p < 0.001$; Fig. 3.5). This lack of flexibility in switching tasks resulted in a high proportion of idle workers with DTM so as to respond efficiently to changes in colony needs (Fig. 3.3). By contrast, with the PTM and ETM the workers readily switched between the regulatory and foraging tasks (all pairwise Wilcoxon tests: $df = 29$, $p < 0.001$; Fig. 3.5), thus not requiring the colonies to maintain a high proportion of idle workers (Fig. 3.3).

The difference in foraging strategies induced by the three threshold models trans-

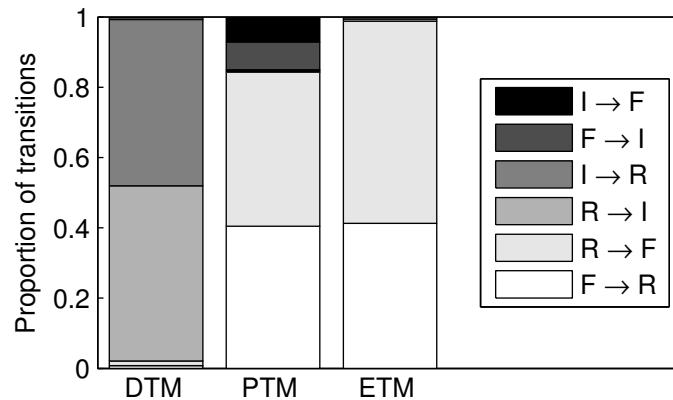


Figure 3.5: Mean proportion of transitions between being engaged in foraging and regulatory tasks and staying idle, summed over 1000 workers. *R* stands for the regulatory task, *F* stands for the foraging task, *I* stands for idle and \rightarrow stands for the direction of the transition. Results are given for each of the three models, over 100 time-steps and 1000 colonies of the 1000th generation (30 replicates). Normalized separately for each model, by the number of all state transitions (number of all state transitions DTM: 3829 ± 147 , PTM: 10926 ± 296 , ETM: 4543 ± 327).

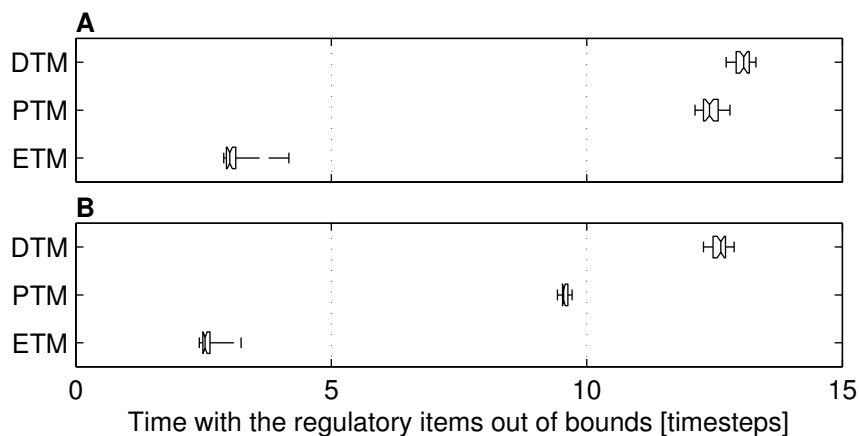


Figure 3.6: Box and Whisker plots showing the number of time-steps: A) that the regulatory items were out of the predefined bounds during entire colony lifespan, B) until the regulatory items were within the predefined bounds for the first time. Results are given for each of the three models, over 1000 colonies of the 1000th generation (30 replicates).

lated in variation in the proportion of time during which the regulatory items were out of the predefined bounds during colony lifespan (Kruskal-Wallis test: $df = 3$, $p < 0.001$). The most efficient regulation was with the ETM (all pairwise Wilcoxon tests: $df = 29$, $p < 0.001$; Fig. 3.6.A). The difference between models in the efficiency of regulation was

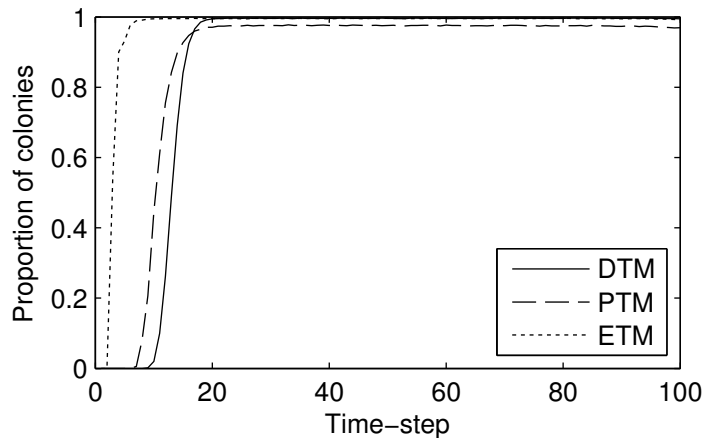


Figure 3.7: Mean proportion of colonies, with regulatory items within the desired bounds as a function of time-step. Results are given for each of the three models, over 1000 colonies of the 1000th generation (30 replicates).

primarily due to two factors. First, there were differences in the time until the regulatory items were within the predefined bounds for the first time (Kruskal-Wallis test: $df = 3$, $p < 0.001$), the most efficient model being the ETM (PTM: +267.5%, DTM: +384.2%; all pairwise Wilcoxon tests: $df = 29$, $p < 0.001$; Fig. 3.6.B). Second, there were differences in the proportion of colonies that successfully kept the regulatory items within the predefined bounds, after they first were within predefined bounds (Kruskal-Wallis test: $df = 3$, $p < 0.001$), the most efficient models being both the DTM and ETM (PTM: -2.1%; Wilcoxon test between DTM and ETM: $df = 29$, $p = 0.11$; all other pairwise Wilcoxon tests: $df = 29$, $p < 0.001$; Fig. 3.7).

We investigated the sensitivity of our findings to changes in the implementation of mutations and in the population's size. We conducted additional experiments with two alternate mutation's implementations (Gaussian mutations and mutations switched off instead of uniform mutations) and two smaller population's sizes (100 and 500 colonies instead of 1000). These experiments showed that the results were robust to changes in both cases (see *Supplementary materials*, Section 3.7.4 for more details).

3.6 Discussion

A comparison of the deterministic (Page Jr and Mitchell 1998), probabilistic (Bonabeau et al. 1996) and our new extended response threshold models showed that they affect the workers' responses to varying stimuli and colony performance. As predicted by our formal analysis, the deterministic response threshold model was found to constrain the

workers' ability to switch tasks because workers with a high foraging threshold and a low regulatory threshold became idle when the regulatory items were within bounds and the foraging stimulus was lower than the workers' thresholds. Consequently, the colony performance was low with the deterministic response threshold model when a dynamic task allocation was required. In the case of the probabilistic response threshold model, the workers' behavioral flexibility was less limited, but the colony was unable to precisely respond to changes in the stimuli values as a response of the stochastic switching between tasks. This led to a relatively low colony performance. Additionally, when the stimuli decreased, the probability that a worker performed a corresponding task decreased too (even if the stimulus was above the threshold). Thus, when the foraging stimulus was low, workers more often switched between being engaged in the foraging task and staying idle, which also had a detrimental effect on colony performance. The extended model did not suffer from these limitations because the weights of the stimuli allowed the workers to switch tasks, while keeping their response deterministic.

We showed that the response threshold models (Bonabeau et al. 1996; Page Jr and Mitchell 1998) and the extended models proposed in this Chapter could be formulated as artificial neural networks. The neuronal formalism introduced here will be useful for further extension of models, such as for example changing the threshold values with age or the integration of adaptive learning, where the connection weights of the neural network are updated using experience-based learning rules (Floreano and Urzelai 2001; Floreano et al. 2008). Furthermore, one could use neural networks with recurrent connections (Mandic and Chambers 2001), to equip the workers with a memory. These, and other, extensions would facilitate addressing increasingly complex and biologically relevant questions on division of labor in social insects. Of note, although we considered a situation with only two tasks, the neuronal formalism can be easily scaled for a higher number of tasks.

We focused on a situation with two tasks, one of which is regulatory (the number of items in the nest ought to be kept within boundaries) and the other that is maximizing (the number of foraged items ought to be maximized). The results allow us to speculate how the models compare in three other situations: 1) with both foraging tasks, 2) with both regulatory tasks and 3) with more than two tasks. First, with two foraging tasks the simulation is strongly simplified, and of little interest, because there is no need to dynamically reallocate the workers between the tasks. Thus, all models should lead to high colony performance. Second, with two regulatory tasks, there could be two outcomes, depending on whether or not a dynamic reallocation of the workers is

required. If not, the workers could split into two distinct subsets and handle the tasks independently. The workers from the first subset would perform the first task, or be idle, and the workers from the second subset would perform the second task, or be idle. In such a case, high performance should be obtained under both the deterministic and extended response thresholds models. While the probabilistic response threshold model, which is unable to provide a precise colony response to a stimulus change, should lead to a lower performance. On the contrary, if the dynamic reallocation of workers is required, the tasks cannot be handled independently by distinct subsets of workers. Consequently, this constitutes similar conditions to those of one regulatory task and one foraging task, and thus the colony performance and behavior should not differ from the ones observed in this study. Finally, if there are more than two tasks, all the limitations of the considered models still hold, and this should not qualitatively affect the results.

Our foraging system implicitly assumed that it was beneficial to minimize the number of idle workers. This might not always be the case in nature, as some idle workers may be beneficial for the colony, for example because of energy constraints (Gordon 1989; Krieger et al. 2000; Robinson 1992) or to serve as reserve force that can be mobilized when needed (Gordon 1989; Robinson 1992; Wilson 1983). It would be possible to include such effects in more complex foraging scenario and study for example the expected relationship between the proportion of idle workers and colony size (Dornhaus et al. 2009; Jeanson et al. 2007). Finally, we assumed that the task stimuli are available to all workers in the colony. What happens to the workers' behavioral flexibility and task allocation efficiency in cases when stimuli are available to only a subset of workers, or depend on spatial configurations, is a question that remains to be investigated.

To further investigate the possible consequences of partial information, we performed a simulation with the version of the deterministic response threshold model proposed by Jeanson et al. (2007). The stimuli are presented to each worker sequentially in a random order, and not all at once, as is assumed by Page Jr and Mitchell (1998). In particular, the workers always perform the first encountered task for which their threshold is lower than the corresponding stimulus. Thus, although the decisions made by the workers are deterministic, the model does not limit the workers' ability to switch tasks, because of the random order of task encounters. The performance and the behavior of the colonies evolved with the deterministic response threshold model with random task encounters are similar to the ones obtained for treatments with the extended response threshold model (*Supplementary materials*, Section 3.7.3). This similarity of the phenotypic traits evolved using very different mechanisms of task allocation is interesting and

might be a promising direction for further studies.

The main focus of this Chapter was to compare several commonly used models of task-allocation. We showed that by allowing for variation not only in stimulus response thresholds, but also by adding the weights to these perceived stimuli, one obtains a much more flexible task allocation system. In contrast to the original response thresholds models, the extended response thresholds model performs well under a wide range of environmental stimuli. However, it remains to be investigated what rule ants and other social insects use. In particular it would be interesting to study whether social insects employ more sophisticated rules than fixed threshold. While evolution is often considered to be an effective optimization process (Parker and Smith 1990) there are many factors such as stochasticity, genetic drift, insufficient time to reach the optimum or the existence of local maxima and other developmental and physiological constraints that may lead to non-optimal behavior (Pérez-Escudero et al. 2009). With social insects the question of evolutionary optimality is especially difficult to address, because colonies are complex, multi-component systems. There are multiple functions on which persistence of the colony depends (e.g., foraging, colony maintenance, defense) and many constraints that the colony must respect (e.g., spatial, energy). Consequently, one cannot rule out the possibility that if a foraging strategy is sub-optimal, this might be due to increase performance of some other tasks that are also important.

It has recently been suggested that in systems of many components, the largest deviations from optimality are expected in those components with less impact on the indirect measure of fitness (Pérez-Escudero et al. 2009). Applying this idea to social groups leads to some predictions associated with colony size differences. For example, an efficient and flexible task allocation is expected to be particularly important in small colonies which typically contain only few idle workers. By contrast, deviations from optimality in task switching might have a lower impact in large colonies which usually contain a substantial reserve force that can be mobilized when needed (Gordon 1989; Robinson 1992; Wilson 1983). Of interest would thus be to compare the mechanisms of task allocation among species varying in colony size and also investigate whether there are differences within species during the ontogeny of the colony.

Overall, our analyses highlight the limitations of the response threshold models that are currently used in the literature (Bertram et al. 2003; Beshers and Fewell 2001; Bonabeau et al. 1996; Graham et al. 2006; Jeanson et al. 2007; Page Jr and Mitchell 1998; Robinson 1987a, 1992; Smith et al. 2008b; Theraulaz et al. 1998). We extended these models by weighting the stimuli. We also showed that the response thresholds model

could be formulated as artificial neural networks thus providing a solid theoretical framework for further studies. Finally, it is worth mentioning that although threshold models have been developed to explain division of labor in social insect, they may also be used to devise efficient systems of task allocation and dynamic scheduling in engineering (Berman et al. 2009; Bonabeau et al. 2000; Campos et al. 2000; Mataric et al. 2003).

3.7 Supplementary materials

3.7.1 Mathematical model of task allocation

We assume that workers receive information of the colony needs via commonly perceived stimuli (Bonabeau et al. 1996; Page Jr and Mitchell 1998) and that the workers' thresholds, and weights, do not vary during their lifetime (Bonabeau et al. 1996; Jeanson et al. 2007; Page Jr and Mitchell 1998). Following are the formal definitions of the deterministic response threshold model (DTM, Page Jr and Mitchell (1998)), the probabilistic response threshold model (PTM), the extended response threshold model (ETM) and the original probabilistic response thresholds model (oPTM) adopted from (Bonabeau et al. 1996).

Let $T = \{1, 2, 3, \dots, n\}$ be the set of all time-steps indices. Let $M = \{1, 2, 3, \dots, m\}$ be the set of all workers indices. Let $s^A(t)$ be the stimulus of task A at the time-step t . Let $s^B(t)$ be the stimulus of task B at the time-step t . For every worker j there are two decision variables $o_j^A(t)$ and $o_j^B(t)$ that translate into being engaged in task A or B , or staying idle.

DTM and PTM: $o_j^A(t) = 1 \cdot s^A(t) - \theta_j^A$ and $o_j^B(t) = 1 \cdot s^B(t) - \theta_j^B$ where θ_j^A and θ_j^B are the thresholds of the worker j for tasks A and B , respectively.

ETM: $o_j^A(t) = w_j^A \cdot s^A(t) - \theta_j^A$ and $o_j^B(t) = w_j^B \cdot s^B(t) - \theta_j^B$ where w_j^A and w_j^B are the weights of the worker j for stimuli A and B , respectively.

oPTM: $o_j^A(t) = \frac{(s^A)^2}{(s^A)^2 + (\theta_j^A)^2}$ and $o_j^B(t) = \frac{(s^B)^2}{(s^B)^2 + (\theta_j^B)^2}$.

Let $W_j(t) \in \{A, B, I\}$ be the state of the worker j at the time-step t , where A and B stand for being engaged in task A and B , respectively, and I stands for staying idle.

DTM, ETM

$$W_j(t) = \begin{cases} A & \text{if } o_j^A(t) > 0 \text{ and } o_j^A(t) > o_j^B(t) \\ B & \text{if } o_j^B(t) > 0 \text{ and } o_j^B(t) > o_j^A(t) \\ Z & \text{if } o_j^A(t) = o_j^B(t) > 0 \\ I & \text{otherwise} \end{cases} \quad (3.1)$$

PTM, oPTM

$$W_j(t) = \begin{cases} A & \text{if } P_j^A(t) = 1 \text{ and } P_j^B(t) = 0 \\ B & \text{if } P_j^A(t) = 0 \text{ and } P_j^B(t) = 1 \\ Z & \text{if } P_j^A(t) = 1 \text{ and } P_j^B(t) = 1 \\ I & \text{otherwise} \end{cases} \quad (3.2)$$

Z is a random variable, which takes value A with probability $\frac{1}{2}$ (otherwise B). $P_j^A(t)$ and $P_j^B(t)$ are random variables, which take value 1 with probability $o_j^A(t)$ and $o_j^B(t)$, respectively (otherwise 0).

3.7.2 Behavioral flexibility in response thresholds models

Let us consider the DTM (see *Supplementary materials*, Section 3.7.1). In practice, the third condition from Equation 3.1 is rarely met, therefore for further analysis the model can be simplified:

$$W_j(t) = \begin{cases} A & \text{if } s^A(t) - \theta_j^A > 0 \text{ and } s^A(t) - \theta_j^A > s^B(t) - \theta_j^B \\ B & \text{if } s^B(t) - \theta_j^B > 0 \text{ and } s^B(t) - \theta_j^B > s^A(t) - \theta_j^A \\ I & \text{otherwise} \end{cases} \quad (3.3)$$

Let us denote by $T_j^A = \{t \in T \mid W_j(t) = A\}$ the set of time-steps in which worker j is engaged in task A and let us denote by $T_j^B = \{t \in T \mid W_j(t) = B\}$ the set of time-steps in which worker j is engaged in task B . From Equation 3.3 it follows:

$$\begin{aligned} \forall j \in M \forall t_j^A \in T_j^A \forall t_j^B \in T_j^B : \\ s^A(t_j^A) - s^B(t_j^A) > \theta_j^A - \theta_j^B \text{ and} \\ s^A(t_j^B) - s^B(t_j^B) < \theta_j^A - \theta_j^B \end{aligned} \quad (3.4)$$

From Equations 3.4 it follows:

$$\begin{aligned} \forall j \in M \forall t_j^A \in T_j^A \forall t_j^B \in T_j^B: \\ s^A(t_j^B) - s^B(t_j^B) < s^A(t_j^A) - s^B(t_j^A) \end{aligned} \quad (3.5)$$

Equation 3.5 holds true if and only if

$$\begin{aligned} \forall j \in M | T_j^B \neq \emptyset \wedge T_j^A \neq \emptyset: \\ \max_{t \in T_j^B} (s^A(t) - s^B(t)) < \min_{t \in T_j^A} (s^A(t) - s^B(t)) \end{aligned} \quad (3.6)$$

Equation 3.6 means that the difference between stimulus for task B and task A ($s^B(t) - s^A(t)$) in moments when a worker is performing task A ($t \in T_j^A$), is always smaller than the analogous difference ($s^B(t) - s^A(t)$) in moments when a worker is performing task B ($t \in T_j^B$). Note that Equation 3.6 holds true 1) for every worker, 2) regardless of the values of the thresholds and 3) regardless of the type of the two tasks (i.e., the tasks do not need to be foraging and regulation). Equation 3.6 limits the task allocation flexibility, because the sets T_j^A and T_j^B must be such that the Equation 3.6 is met. In other words, regardless of the individual thresholds, the values of stimuli may by definition constrain the workers' ability to switch tasks by restraining the T_j^A set and/or the T_j^B set (and thus restraining the time when the worker j can perform tasks A and B). In the extreme case, if one of the sets T_j^B and T_j^A is empty, then Equation 3.6 is always met, regardless of the other set. Note that if T_j^B or T_j^A is empty, it means a worker j is a specialist (performs only one task or is idle). This explains the experimental results where the number of transitions between performing a regulatory task and staying idle (which corresponds to the number of specialists) was higher while evolving the colonies for treatments with the DTM, than with other treatments (see Section 3.5).

We obtained an analogous constraint (compare Equations 3.6 and 3.7) for the ETM by applying the same reasoning as for the DTM:

$$\begin{aligned} \forall j \in M | T_j^B \neq \emptyset \wedge T_j^A \neq \emptyset: \\ \max_{t \in T_j^B} (w_j^A \cdot s^A(t) - w_j^B \cdot s^B(t)) < \min_{t \in T_j^A} (w_j^A \cdot s^A(t) - w_j^B \cdot s^B(t)) \end{aligned} \quad (3.7)$$

The additional parameters in Equation 3.7 (the weights w_j^B and w_j^A) lift the constraints on the flexibility of task allocation. Every worker has its own set of weights, which can evolve as such, that Equation 3.7 is met even if both T_j^B and T_j^A are not empty. Note that if both sets T_j^B and T_j^A are not empty, a worker j is a generalist (switches tasks). This

explains the experimental results where the number of transitions between performing distinct tasks (which corresponds to the number of generalists) is higher while evolving the colonies for treatments with the ETM, compared to the DTM (see Section 3.5).

The PTM escapes this analysis, because the model is not deterministic and basis on probabilities.

3.7.3 Analyses of other response thresholds models

Deterministic response threshold model with non-negative values of thresholds

Although threshold values are typically expected to be non-negative (Bonabeau et al. 1996; Graham et al. 2006; Jeanson et al. 2007; Page Jr and Mitchell 1998), in our experiments we allowed values between -1 to $+1$ in order to be consistent with the formalism of neural networks weights. We conducted a control experiment to confirm that our choice did not affect the results of the experiments. We found support for this claim, showing that there is a reasonable good agreement between the DTMs with different range of threshold values with respect to colony performance (Fig. 3.8.A) and colony dynamics (Fig. 3.9.AB). In treatment with the DTM $[-1,1]$ each worker had a genome consisting of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$). In treatment with the DTM $[0,1]$ each worker had a genome consisting of two thresholds, both ranging from 0 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{256}$). Other settings were the same as in the main experiments (see Section 3.4).

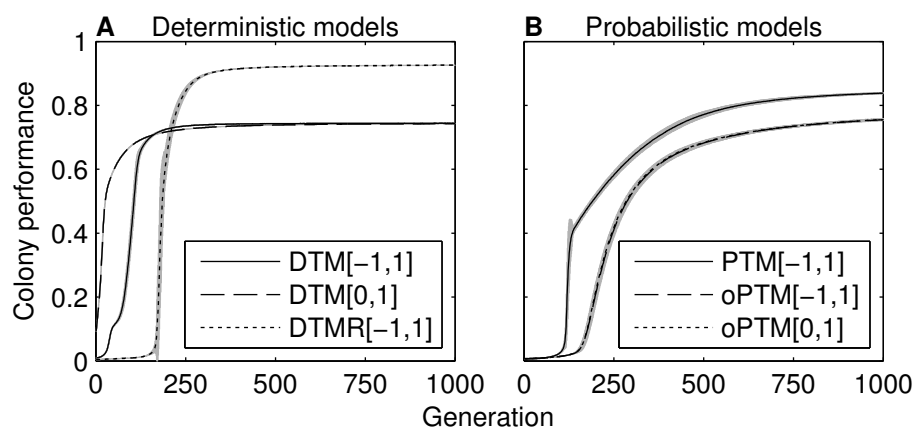


Figure 3.8: Mean \pm s.d. (in grey) performance for three deterministic (A) and three probabilistic (B) response threshold models over 1000 colonies (30 replicates). Note that values of oPTM $[-1,1]$ and oPTM $[0,1]$ match, making the corresponding plots indistinguishable.

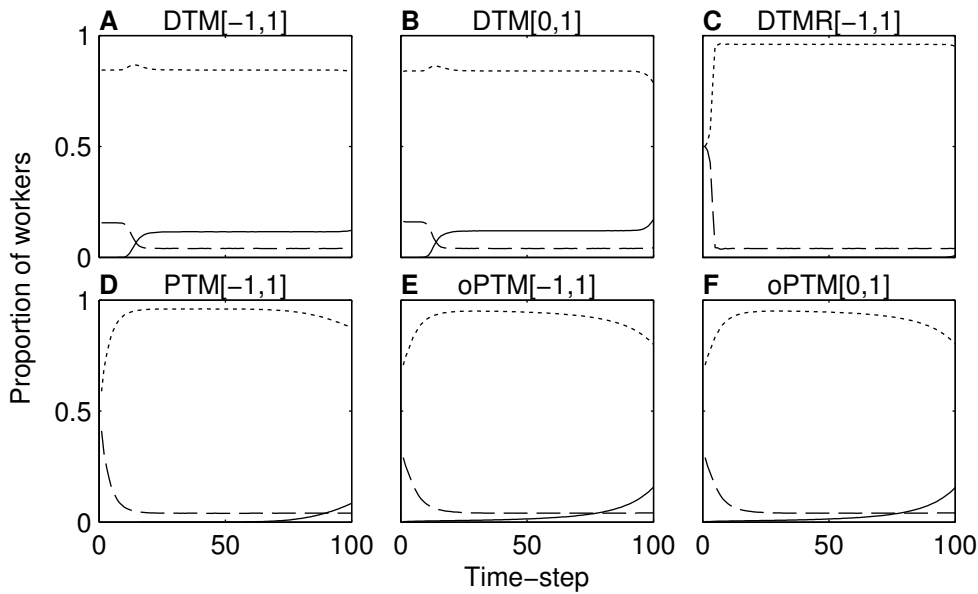


Figure 3.9: Mean proportion of workers engaged in the foraging task (short-dashed line), in the regulating task (long-dashed line) or staying idle (solid line) as a function of time-steps. Results for three deterministic (ABC) and three probabilistic (DEF) response threshold models, over 1000 colonies of the 1000th generation (30 replicates).

Original probabilistic response threshold model

We conducted a control experiment to test whether the PTM (which is in fact a probabilistic interpretation of the deterministic response threshold model proposed by Page Jr and Mitchell (1998)) produces a similar colony performance and workers' behavior, as the original probabilistic response threshold model proposed by Bonabeau et al (Bonabeau et al. 1996) (oPTM). We found support for this claim, showing that there is a reasonable good agreement between different probabilistic response thresholds models with respect to the colony performance (Fig. 3.8.B) and colony dynamics (Fig. 3.9.EFG). In treatments with the PTM[-1,1] and oPTM[-1,1] each worker had a genome consisting of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$). In treatment with the oPTM[0,1] each worker had a genome consisting of two thresholds, both ranging from 0 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{256}$). Other settings were the same as in the main experiments (see Section 3.4).

We tested the oPTM with non-negative values of the thresholds, because in this way the model was defined originally Bonabeau et al. (1996). Of note, on contrary to the models based on artificial neural networks proposed in Section 3.4.1, using negative

thresholds with the oPTM does not increase the explanatory power of the model. This is because in the oPTM the thresholds are risen to the second power and thus information about the sign of the thresholds is disregarded.

Deterministic response threshold model with random order of task encounters

We conducted a control experiment to test the effects of using the DTM as formulated by Jeanson et al. (2007) (DTMR). The stimuli are presented to each workers sequentially in a random order, and not all at once, as in the DTM. In particular, the workers always perform the first encountered task for which their threshold is lower than the corresponding stimulus. Thus, although the decisions made by the workers are deterministic, the model does not limit the workers' ability to switch tasks, because of the stochastic order of task encounters. Interestingly, there is a very good agreement between the DTMR and ETM with respect to the colony performance (compare Figures 3.2 and 3.8.A) and dynamics of the system (compare Figures 3.4 and 3.9.C). It seems that the DTMR and ETM, which are different with respect to the mechanisms of task allocation, lead to the evolution of similar phenotypic traits. The possible implications of this finding still require to be solidly investigated.

In treatment with the DTMR[-1,1] each worker had a genome consisting of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$). Other settings were the same as in the main experiments (see Section 3.4).

3.7.4 Effects of population size and mutation

Effects of population sizes

To test the sensitivity of the results to different population size we performed two control experiments, with populations consisting of 100 colonies and with populations consisting of 500 colonies. Other settings were the same as in the main experiments (see Section 3.4). In particular, we used the same mutation (i.e., with a probability of 0.001 each allele was set randomly to one of the 256 values between -1 and $+1$ with a resolution of $\frac{1}{128}$), to which we refer as uniform mutation. We found that both control experiments are in agreement with the results reported in the main text, with respect to the colony performance (Fig. 3.10.AB) and colony dynamics (Fig. 3.11.A-F).

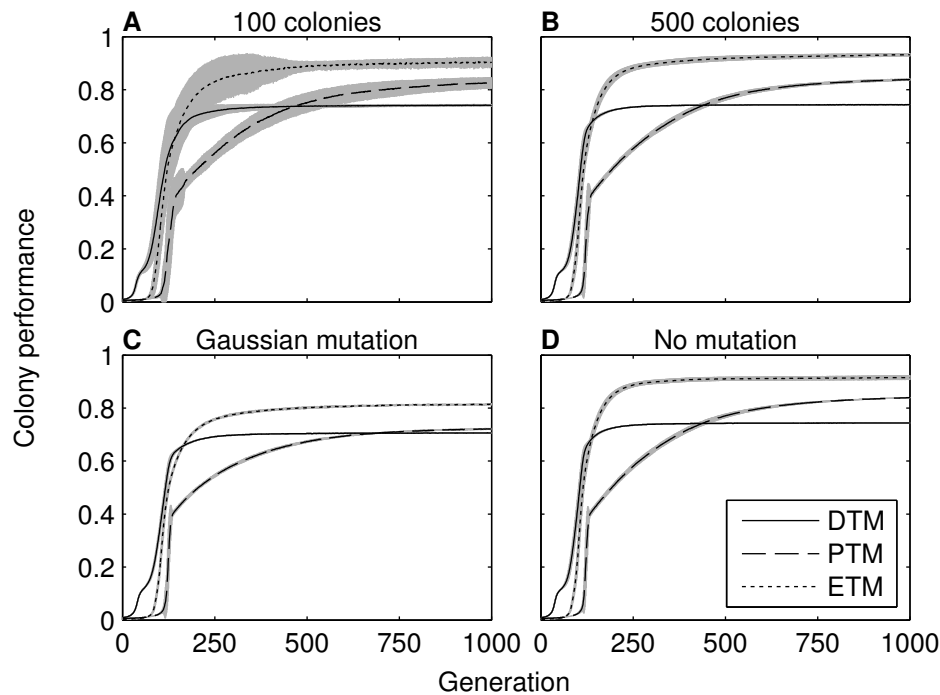


Figure 3.10: Mean \pm s.d. (in grey) performance of each of the three response threshold models (30 replicates). A) 100 colonies were evolved with uniform mutations. B) 500 colonies were evolved with uniform mutations. C) 1000 colonies were evolved with frequent Gaussian mutations. D) 1000 colonies were evolved with mutations switched off.

Effects of mutation

To test the sensitivity of the results to different mutations we performed two control experiments. In the first one, the mutation was switched off. In the second one, all alleles of the workers were mutated by adding a random value with a resolution of $\frac{1}{128}$. The values of mutations were generated using a Gaussian distribution (mean \pm s.d.: 0 ± 0.0075), so that the magnitude of the value was inversely proportional to its probability of occurrence. We refer to this mutation as frequent Gaussian mutation. All alleles that by mutation became lower than -1 or greater than $+1$ were reflected back above -1 or below $+1$ accordingly (i.e., if the value after mutation was $v > +1$ we used $2 - v$ instead, and if the value after mutation was $v < -1$ we used $-2 - v$ instead). Other settings were the same as in the main experiments (see Section 3.4). We found that both control experiments are in agreement with the results reported in the main text, with respect to the colony performance (Fig. 3.10.CD) and colony dynamics (Fig. 3.11.G-L)

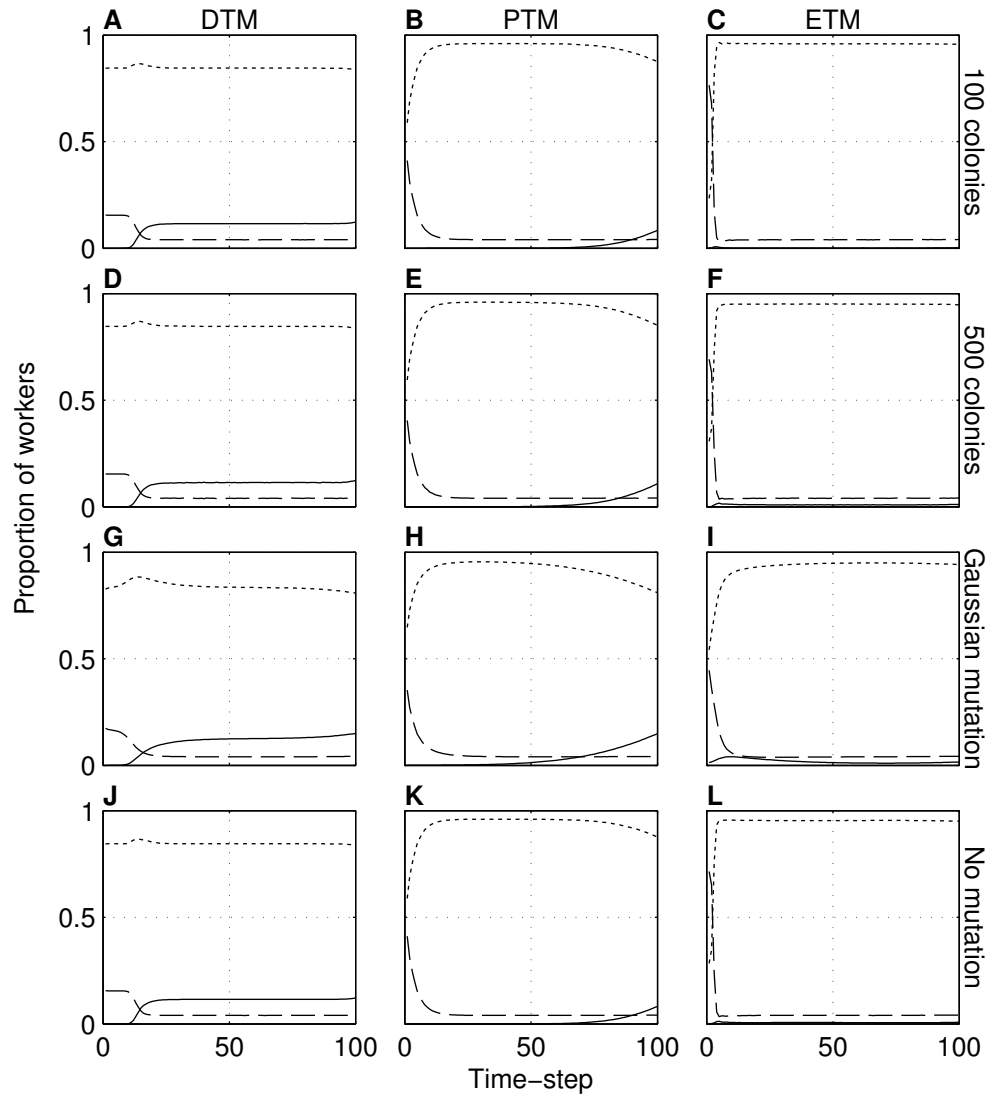
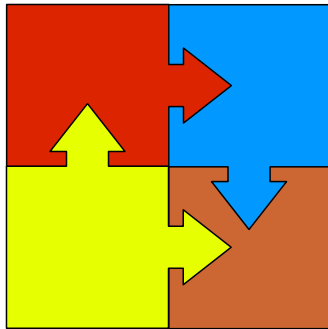


Figure 3.11: Mean proportion of workers engaged in the foraging task (short-dashed line), in the regulating task (long-dashed line) or staying idle (solid line) as a function of time-steps. Results for three response threshold models, over all colonies of the 1000th generation (30 replicates). 100 colonies were evolved with uniform mutations and with the deterministic (A), probabilistic (B), and extended (C) response threshold model. 500 colonies were evolved with uniform mutations and with the deterministic (D), probabilistic (E), and extended (F) response threshold model. 1000 colonies were evolved with frequent Gaussian mutations and with the deterministic (G), probabilistic (H), and extended (I) response threshold model. 1000 colonies were evolved with mutations switched off and with the deterministic (J), probabilistic (K), and extended (L) response threshold model.

Conclusions



In this thesis, we demonstrated how evolutionary simulations performed in a computer contribute to biological understanding and ease the engineering of multi-agent systems dividing the labor among its members. Throughout the three chapters we used increasingly complex models, each time taking a different perspective on evolution of specialization, as well as focusing on a different methodological aspect. In Chapter 1, we investigated the origins of cooperation, a prerequisite of division of labor, and the role played by the selection methods (mechanisms of intra-specific selection). Next, in Chapter 2, we studied the evolution of division of labor at the genetic level, and the role played by crossover operators. Finally, in Chapter 3, we looked at specialization at the behavioral level, and investigated the mechanisms of individual decision making. In this concluding remarks we summarize our main contributions and point to interesting directions of future research related to the evolution of division of labor.

Main contributions

We investigated the transition from selfishness to cooperation in societies of agents which behavior was reduced to a binary, genetically encoded choice between cooperation and defection (see Chapter 1). We supported the long recognized view that the relatedness level and the costs and benefits of cooperation are important factors influencing the evolution of cooperation (Hamilton 1964). However, we showed theoretically and experimentally that these factors are regulated by the mechanisms of intra-specific selection. Consequently, the selection methods (that in evolutionary simulations model differential reproduction of individuals) affect the conditions where the cooperative behavior spreads. This result broadened our understanding of the origins of cooperation and bridged the gap between mathematical models and computer simulations used to study the evolution of social traits.

Once we learnt when cooperation evolves, we moved forward and studied the evolution of fixed assignments of cooperative agents to tasks (see Chapter 2). In comparison to Chapter 1, we increased the behavioral repertoire of agents. Each agent took one of many genetically encoded roles, but only some of them and in some compositions were beneficial for the team. We studied the effects of two team crossover operators which exchanged entire agents between the teams, either keeping or destroying the ordering of agents within the teams. We proposed a combination of the two operators which balanced the exploration and exploitation aspect of the evolutionary simulation and, consequently, found globally optimal team compositions. We, thus, contributed novel algorithms to the field of team optimization and provided a thorough analysis of team crossover.

Finally, knowing how to evolve fixed specialization, we took one more step and employed ourselves with the evolution of flexible task allocation mechanisms in simulated ant colonies (Chapter 3). In comparison to Chapter 2, we further increased the complexity of agents by equipping them with simple artificial neural networks, which mapped stimuli from the environment into the decision which task to perform. Within this neuronal formalism, we expressed several biological models used to explain division of labor in social insects and compared their efficiency in task allocation by means of evolutionary simulations. We showed that the models failed to explain optimal colony performance when precise and quick reallocation of the workforce was required. We overcame this issue by taking advantage of the neuronal formalism which we had introduced, i.e., by extending the models with additional weights of the neuronal connections. This result proposed new techniques to model and gave interesting insights on collective

dynamics of task allocation in groups of autonomous agents.

We investigated full evolutionary pathway for division of labor, from the occurrence of cooperation between competitive agents, through specialization fixed at a genetic level, up to task allocation mechanisms allowing the behavioral flexibility. In parallel, we studied a spectrum of methodological issues related to the evolution and optimization of multi-agent teams dividing the labor. We presented the method of evolutionary simulations and its applications in biology to study social traits (Chapter 1). We systemized evolutionary approaches used by engineers to optimize team behavior (Chapter 2). We studied in details the effects of the selection methods and the crossover operators on the evolutionary dynamics of cooperation (Chapter 1) and fixed specialization (Chapter 2). Finally, we provided a comprehensive framework to model task allocation with artificial neural networks, which will be useful for further extension like the integration of adaptive learning (Chapter 3).

Overall, we demonstrated that evolutionary simulations are promising tools to address biological questions, as well as to engineer multiple autonomous agents, thus allowing for beneficial synergies between the fields. Nowadays, scientists study and design multi-agent systems of such level of complexity that they are beyond the reach of any single methodology, let it be formal analyses, wet-lab and field experiments, or even human intuition. The middle ground in the form of simulations in a computer that allow to control every aspect of the model, and yet to make it arbitrarily complex (e.g., Karr et al. 2012), appear to be the future of studies on complex systems (e.g., Adami 2012).

Future directions

Looking at the broad spectrum of this thesis, there are various directions for future research, many of which we have already presented in the *Discussion* sections of the three main chapters (Sections 1.6, 2.6, 3.6). Here, we will not repeat the points already made. Instead, we would like to synthesize them, as to broadly sketch a few main challenges related to evolutionary simulations of multi-agent systems in engineering and biology.

In general, evolutionary simulations are above all a useful tool of optimization (Fogel 1994). We support this view by using them to efficiently find desirable compositions of agents within teams (Chapter 2). With this respect, we have considered relatively simple agents. In contrast, other authors evolved more complex agents, but assuming a

Conclusions

fixed team composition (see Panait and Luke 2005 for a review). The big challenge in engineering sciences is to merge these two paths and propose efficient techniques for simultaneous evolution of agents and of teams. Our work seems to provide the missing puzzle, and perhaps will open the doors to address this issue in near future.

Moreover, many engineering studies introducing novel algorithms often state only how good the method is on the selected benchmarks, and unfortunately do not explain why it is so good. Consequently, the link between the novel approach and specific problems for which it should be efficient often remains unclear, and the authors' work fails to provide a long-lasting valuable contribution. To go against this trend, we not only proposed a novel way to evolve agent compositions within teams, but we also explained the conditions when and the reasons why it was efficient (Chapter 2). To address this issue in a broader perspective, a "hyper-heuristic" scheme was proposed in which a proper optimization algorithm is chosen and fine-tuned automatically (Özcan et al. 2008). However, hyper-heuristics are still in their infancy and to our knowledge they have not yet been applied in team optimization, which appears to be a very interesting research direction.

From a biological perspective, we provided interesting insights on the evolution of cooperation and division of labor by means of evolutionary simulations mixed with formal analyses. However, biology is ultimately the study of life, and not of mathematical models, or computer simulations. Thus, in biology, the big challenge is to validate theoretical (mathematical or simulated) predictions against real world data. Unfortunately, such data is often missing. For example, with respect to the selection methods (Chapter 1), it would be valuable to measure fitness of species in the wild (e.g., Rodriguez-Munoz et al. 2010), and to quantify the inter-individual interactions leading to death and reproduction in real populations (e.g., Endler 1986). With respect to task allocation in social insects (Chapter 3), one could investigate the optimality of the observed patterns of division of labor, which might require careful considerations of workers' metabolism (e.g., Schmid-Hempel 1990; Schmid-Hempel and Wolf 1988). It has been argued that without the validation against real data, many modeling efforts provide only "smoke and mirrors" instead of deep understanding of biological processes (Gordon 2007). Our observations seem to somehow support this view, as we exposed that little details are often crucial for the evolutionary (Chapter 1) and collective (Chapter 3) dynamics of the studied systems. Nevertheless, simulations of artificial organisms remain a rich source of inspiration, and when eventually combined with behavioral data, we believe they will allow a revolution similar to the one that took place in genomics.

A Association between performance and fitness in evolutionary simulations

To illustrate the differences between the selection methods we simulated the creation of the descending generation. We considered a population of 1000 individuals with fixed performance scores following normal distribution with mean 0.5 and standard deviation 0.125. There were 11 treatments: PSM, RSM, TPSM with truncation threshold $t = 0.8, 0.5, 0.2$, TUSM with truncation threshold $t = 0.8, 0.5, 0.2$, TSM with tournament size $s = 2, 3, 5$. To determine the relationship between performance scores and fitness, we calculated for each individual the mean \pm s.d. number of offspring in the descending generation (over the 10 000 replicates).

Each selection method imposed a different mapping from performance to fitness (Fig. A.1). The number of offspring that an individual contributed on average to the descending generation was linearly proportional to its performance only with PSM. In contrast, with RSM, TPSM, TUSM, and TSM the relationship between performance and fitness was non-linear. Moreover, with PSM, RSM, and TSM, the relationship was continuous, in contrast to TPSM and TUSM for which the truncation introduced discontinuity. Finally, the strength of the selection pressure increased with the decrease of the truncation threshold t under TPSM and TUSM, and with the increase of the tournament size s under TSM. In particular, individuals with high performance scores contributed the most offspring to the descending generation with the lowest truncation threshold ($t = 0.2$) and the highest tournament size ($s = 5$). Similarly, individuals with low performance scores contributed the least offspring to the descending generation with the highest truncation threshold ($t = 0.8$) and with the lowest tournament size ($s = 2$).

Appendix A. Association between performance and fitness in evolutionary simulations

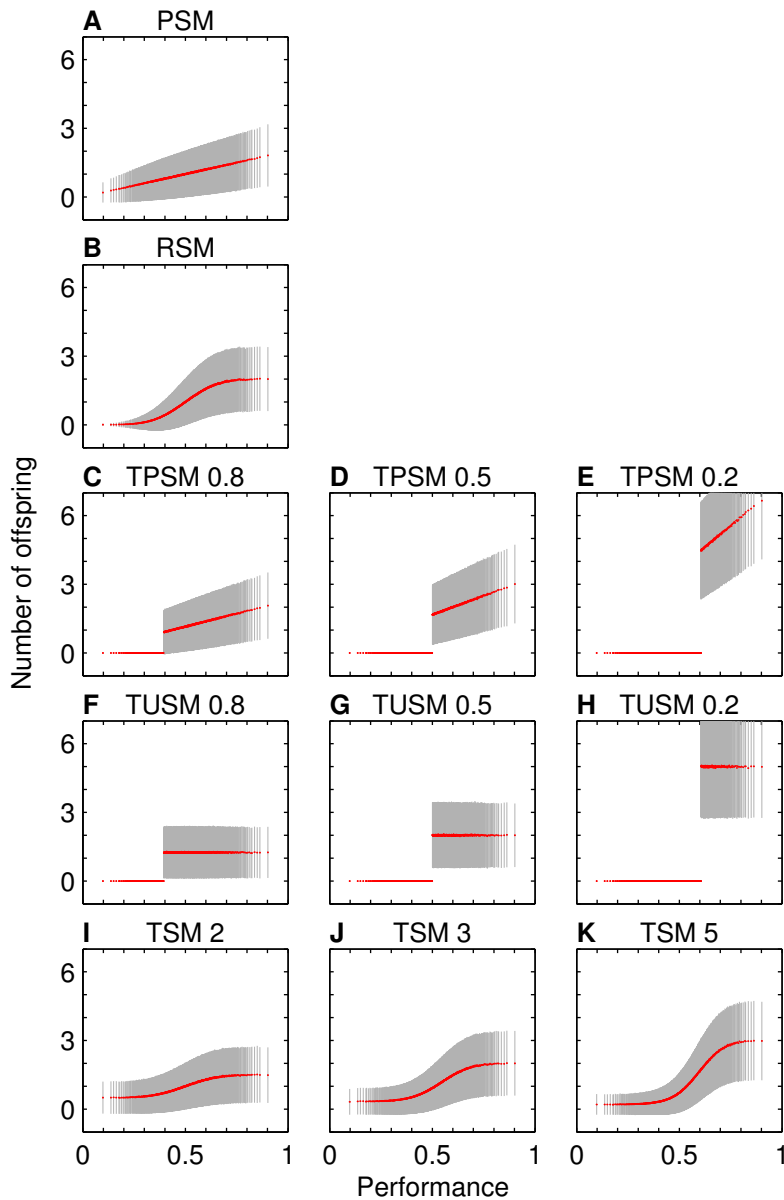


Figure A.1: Mean \pm s.d. (in gray) number of offspring that individuals contribute to the descending generation versus their performance scores (10 000 replicates). There were 11 treatments, and in each a different selection method was used: proportionate (A), rank (B), truncation-proportionate with threshold $t = 0.8$ (C), $t = 0.5$ (D), and $t = 0.2$ (E), truncation uniform with threshold $t = 0.8$ (F), $t = 0.5$ (G), and $t = 0.2$ (H), and tournament with size $s = 2$ (I), $s = 3$ (J), and $s = 5$ (K). Population contained 1000 individuals, each having a fixed performance scores drawn from a normal distribution with mean 0.5 and standard deviation 0.125.

This result highlights the important difference between performance and fitness. Performance is a measure of an individual's phenotype, whereas fitness is a measure of its reproductive success. With this respect, there exists a terminological ambiguity that perpetuates the literature on evolutionary simulations. With PSM it is legitimate to refer to performance as fitness, because relative fitness is irrelevant of scaling Wagner (2010). But historically, performance of individuals has been called fitness regardless of the selection method applied (see, e.g., Goldberg 1989; Mitchell 1996). In the light of the differences between the two concepts laid down in Chapter 1 this seems unfortunate. The naming convention is not crucial in engineering as it does not affect the optimization process, neither its outcome which is in the focus of practical applications. However, in biological studies miscalling performance as fitness does not conform to the rich body of work in evolutionary theory (see, e.g., Rice 2004) and might introduce unnecessary confusion into the field.

B Application of team crossover operators in the evolution of decentralized controllers for task allocation

We validated the restricted agent swapping (RAS) and free agent swapping (FAS) in a complex problem of evolving decentralized controllers for task allocation. To this aim we quantified the teams' efficiency in task allocation using a stochastic agent-based simulation developed in Chapter 3 (see also Appendix C). We considered a team composed of 1000 agents which allocated themselves to two distinct tasks using the deterministic response threshold model (DTM) and the extended response thresholds models (ETM) (see Section 3.7.1 for the explanation of the models)

In experiments with DTM and ETM, each team had a genome consisting of 1000 parts, which corresponded to the agents. With DTM, each of these 1000 genome's parts consisted of two thresholds, both ranging from -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$). With ETM, each of the 1000 genome's parts consisted of two thresholds and two weights, all ranging from -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$).

We performed 2000 generations of artificial selection in thirty independent replicates for each of the two models (DTM and ETM), for two treatments (RAS and FAS), and in two conditions (without and with mutations). Overall there were $2 \times 2 \times 2 = 8$ experimental lines. In the experiments with mutations each allele of the newly added teams to the population was randomly set to a value between -1 to $+1$ (8-bit encoding, 256 possible values with a resolution of $\frac{1}{128}$) with a probability 0.001. Other settings were the same as in Chapter 2 (see Section 2.4).

Appendix B. Application of team crossover operators in the evolution of decentralized controllers for task allocation

There were important differences in team performance between the two treatments (RAS and FAS), under both conditions (without and with mutations), and for both models (DTM and ETM). The team performance was higher with FAS than with RAS for both models, when mutations were not used (Fig. B.1.AB; mean performance \pm s.d. at generation 2000 with DTM and without mutations, FAS: $(7438 \pm 6) \cdot 10^{-4}$, RAS: $(6698 \pm 141) \cdot 10^{-4}$; with ETM and without mutations, FAS: $(9161 \pm 9) \cdot 10^{-4}$, RAS: $(6695 \pm 225) \cdot 10^{-4}$; both Wilcoxon tests, $df = 29$, $p < 0.001$). With DTM and with mutations, there was a significant difference in performance between the two treatments, however it was very small (Fig. B.1C; mean performance \pm s.d. at generation 2000, FAS: $(7304 \pm 7) \cdot 10^{-4}$, RAS: $(7298 \pm 10) \cdot 10^{-4}$; Wilcoxon test, $df = 29$, $p < 0.01$). With ETM and with mutations, the performance was higher with FAS than with RAS (Fig. B.1D; mean performance \pm s.d. at generation 2000, FAS: $(8966 \pm 17) \cdot 10^{-4}$, RAS: $(8672 \pm 22) \cdot 10^{-4}$; Wilcoxon test, $df = 29$, $p < 0.001$). The results confirm our observations that using mutations with RAS may to some extent overcome premature convergence (Section 2.7.3). Nevertheless, for the evolution of more complex agents (here ETM) FAS remained superior to RAS. Moreover, in all cases FAS led to a faster evolution of teams displaying the highest performance (Fig. B.1). Note that the performance differences between the DTM and ETM are due to limitations in task switching implicitly present in the formulation of the DTM (see Chapter 3 for more details).

Note that, in the scenario used here there are two tasks, thus the optimal team consists probably of a few large groups of agents of the same type, e.g., regulators and foragers. This explains the observed higher performance in treatments with FAS than with RAS (in the main experiment we showed that using FAS is a more efficient strategy of evolving team compositions when the optimal team has a rather low level of agent heterogeneity). Thus, we believe that the results presented in this Appendix support our conclusions from Chapter 2 and show that they are also valid for more complex problems.

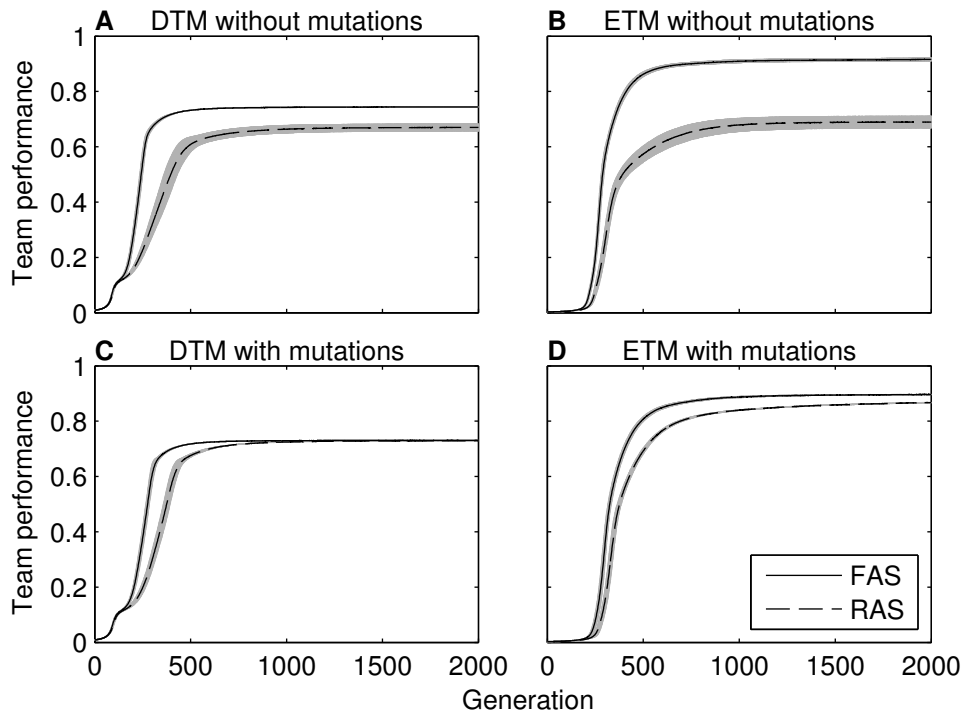


Figure B.1: Mean performance (\pm s.d. in grey) of 1000 teams evolved with restricted agent swapping (blue) and free agent swapping (red) for two treatments: (a) deterministic response thresholds model and (b) extended response thresholds model (30 replicates). Teams consisted of 1000 agents, which needed to dynamically self-allocate to two different tasks in order for the team to achieve the optimal performance.

C Formal definition of the task allocation simulation

The stochastic simulation presented in this Appendix was used extensively in Chapter 3 to investigate biological models of division of labor in social insects, and in Appendix B (related to Chapter 2) to investigate team crossover operators in the evolution of task allocation in teams of artificial agents.

Let $T = \{1, 2, 3, \dots, n\}$ be the set of all time-steps indices ($n = 100$). Let $M = \{1, 2, 3, \dots, m\}$ be the set of all workers (agents) indices ($m = 1000$). The state of each worker (agent) j in every time-step t ($W_j(t)$) is decided according to the chosen task allocation model (see Section 3.7.1), where task A corresponds to foraging task F and task B corresponds to regulatory task R . The foraging stimulus $s^F(t)$ and the regulatory stimulus $s^R(t)$ are defined as

$$s^F(t) = 1 - 0.0001 \cdot a^F(t-1)$$

$$s^R(t) = 1 - 0.005 \cdot a^R(t-1)$$

$a^F(t)$ and $a^R(t)$ are the number of items accumulated in the nest (base) at the time-step t , and are defined as

$$a^F(0) = 0$$

$$a^R(0) = 0$$

$$a^F(t) = \sum_{i=1}^t (g^F(i) - d^F(i))$$

$$a^R(t) = \sum_{i=1}^t (g^R(i) - d^R(i))$$

Appendix C. Formal definition of the task allocation simulation

$g^F(t)$ and $g^R(t)$ are the number of items gathered at the time-step t , and are defined as

$$g^F(t) = \sum_{j=1}^m X_j^F(t)$$

$$g^R(t) = \sum_{j=1}^m X_j^R(t)$$

$X_j^F(t)$ and $X_j^R(t)$ are random variables such that

$$X_j^F(t) = \begin{cases} 1 & \text{with probability } p^F \text{ if } W_j(t) = F \\ 0 & \text{otherwise} \end{cases}$$

and

$$X_j^R(t) = \begin{cases} 1 & \text{with probability } p^R \text{ if } W_j(t) = R \\ 0 & \text{otherwise} \end{cases}$$

The probability of successful foraging p^F is set to value 0.1. The probability of successful regulation p^R is set to value 0.1. Values $d^F(t)$ and $d^R(t)$ are the number of items depleted from the nest (base) at the time-step t , and are defined as

$$d^F(t) = \begin{cases} 0 & \text{if } d^F(t-1) = 10 \\ D & \text{if } d^F(t-1) = 0 \text{ and } d^F(t-2) = 10 \\ 10 & \text{if } d^F(t-1) = 0 \text{ and } d^F(t-2) = 0 \end{cases}$$

and

$$d^R(t) = \begin{cases} 0 & \text{if } d^R(t-1) = 10 \\ D & \text{if } d^R(t-1) = 0 \text{ and } d^R(t-2) = 10 \\ 10 & \text{if } d^R(t-1) = 0 \text{ and } d^R(t-2) = 0 \end{cases}$$

where D is a random variable, which takes value 10 with probability $\frac{1}{2}$ (otherwise 0).

Bibliography

- C. Adami. Digital genetics: unravelling the genetic basis of evolution. *Nature Reviews Genetics*, 7(2):109–118, 2006.
- C. Adami. Boldly going beyond mathematics. *Science*, 338(6113):1421–1422, 2012.
- F.R. Adler and D.M. Gordon. Information collection and spread by networks of patrolling ants. *The American Naturalist*, 140(3):373–400, 1992.
- A. Agah and G.A. Bekey. Phylogenetic and ontogenetic learning in a colony of interacting robots. *Autonomous Robots*, 4(1):85–100, 1997.
- A. Agah and K. Tanie. Robots playing to win: evolutionary soccer strategies. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1997)*, pages 632–637. IEEE, 1997.
- W. Agassounon and A. Martinoli. Efficiency and robustness of threshold-based distributed allocation algorithms in multi-agent systems. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 3*, pages 1090–1097. ACM, 2002.
- A. Agogino, K. Stanley, and R. Miikkulainen. Online interactive neuro-evolution. *Neural Processing Letters*, 11(1):29–38, 2000.
- D. Andre and A. Teller. Evolving team darwin united. In M. Asada and H. Kitano, editors, *RoboCup-98: Robot Soccer World Cup II*, volume 1604 of *Lecture Notes on Computer Science*, pages 346–351. Springer Verlag, 1999.
- R. Axelrod. The evolution of strategies in the iterated prisoner’s dilemma. In L. Davis, editor, *Genetic Algorithm and Simulated Annealing*, pages 32–41, London, 1987. Pitman.
- T. Back. Selective pressure in evolutionary algorithms: A characterization of selection mechanisms. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, pages 57–62. IEEE, 1994.

Bibliography

- D.A. Bader, V. Agarwal, and S. Kang. Computing discrete transforms on the cell broadband engine. *Parallel Computing*, 35(3):119–137, 2009.
- T. Balch and R.C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.
- G. Baldassarre, S. Nolfi, and D. Parisi. Evolving mobile robots able to display collective behaviors. *Artificial Life*, 9(3):255–267, 2003.
- S. Berman, A. Halasz, M.A. Hsieh, and V. Kumar. Optimized stochastic policies for task allocation in swarms of robots. *IEEE Transactions on Robotics*, 25(4):927–937, 2009.
- S.M. Bertram, R. Gorelick, and J.H. Fewell. Colony response to graded resource changes: an analytical model of the influence of genotype, environment, and dominance. *Theor Popul Biol*, 64(2):151–162, 2003.
- S.N. Beshers and J.H. Fewell. Models of Division of Labor in Social Insects. *Annual Review of Entomology*, 46(1):413–440, 2001.
- S.N. Beshers, Z.Y. Huang, Y. Oono, and G.E. Robinson. Social inhibition and the regulation of temporal polyethism in honey bees. *Journal of theoretical biology*, 213(3):461–479, 2001.
- R. Bianco and S. Nolfi. Toward open-ended evolutionary robotics: evolving elementary robotic units able to self-assemble and self-reproduce. *Connection Science*, 16(4):227–248, 2004.
- T. Blickle and L. Thiele. A mathematical analysis of tournament selection. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 9–16. Morgan Kaufmann Publishers Inc., 1995.
- E. Bonabeau, G. Theraulaz, and J.L. Deneubourg. Quantitative study of the fixed threshold model for the regulation of division of labour in insect societies. *Proceedings of the Royal Society B: Biological Sciences*, 263:1565–1569, 1996.
- E. Bonabeau, A. Sobkowski, G. Theraulaz, and J.L. Deneubourg. Adaptive task allocation inspired by a model of division of labor in social insects. In D. Lundh, B. Olsson, and A. Narayanan, editors, *Bio Computation and Emergent Computing*, pages 36–45, 1997a.
- E. Bonabeau, G. Theraulaz, J.L. Deneubourg, S. Aron, and S. Camazine. Self-organization in social insects. *Trends in Ecology & Evolution*, 12(5):188–193, 1997b.
- E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm intelligence: from natural to artificial systems*. Oxford University Press, New York, 1999.

- E. Bonabeau, M. Dorigo, and G. Theraulaz. Inspiration for optimization from social insect behaviour. *Nature*, 406(6791):39–42, 2000.
- J.C. Bongard. The legion system: A novel approach to evolving heterogeneity for collective problem solving. In R. Poli, W. Banzhaf, Langdon W.B., J. Miller, P. Nordin, and Fogarty T.C., editors, *Proceedings of the European Conference on Genetic Programming (EuroGP 2000)*, volume 1802 of *Lecture Notes on Computer Science*, pages 16–28. Springer-Verlag, 2000a.
- J.C. Bongard. Reducing collective behavioural complexity through heterogeneity. In *Artificial Life VII: Proceedings of the Seventh International Conference*, pages 327–336. The MIT Press, 2000b.
- H.M. Botee and E. Bonabeau. Evolving ant colony optimization. *Complex Systems*, 1(2): 149–159, 1998.
- A.F.G. Bourke. Colony size, social complexity and reproductive conflict in social insects. *Journal of Evolutionary Biology*, 12(2):245–257, 1999.
- S. Bowles and H. Gintis. The evolution of strong reciprocity: cooperation in heterogeneous populations. *Theoretical population biology*, 65(1):17–28, 2004.
- M. Brameier and W. Banzhaf. Evolving teams of predictors with linear genetic programming. *Genetic Programming and Evolvable Machines*, 2(4):381–407, 2001.
- R.A. Brooks. New approaches to robotics. *Science*, 253(5025):1227, 1991a.
- R.A. Brooks. Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159, 1991b.
- L. Bull. Evolutionary computing in multi-agent environments: Operators. In V.W. Porto, N. Saravanan, D.E. Waagen, and A.E. Eiben, editors, *Proceedings of the Seventh International Conference on Evolutionary Programming (EP98)*, volume 1447 of *Lecture Notes in Computer Science*, pages 370–377. Springer, 1998.
- M. Campos, E. Bonabeau, G. Theraulaz, and J.L. Deneubourg. Dynamic scheduling and division of labor in social insects. *Adaptive Behavior*, 8(2):83–95, 2000.
- A. Cangelosi. Evolution of communication and language using signals, symbols, and words. *Evolutionary Computation, IEEE Transactions on*, 5(2):93–101, 2001.
- A. Cangelosi and D. Parisi. The emergence of a ‘language’ in an evolving population of neural networks. *Connection Science*, 10(2):83–97, 1998.

Bibliography

- J. Cao. Self-organizing agents for grid load balancing. In *Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing*, pages 388–395. IEEE, 2004.
- A.M. Colman. The puzzle of cooperation. *Nature*, 440(7085):744–745, 2006.
- J.F. Crow and M. Kimura. Efficiency of truncation selection. *Proceedings of the National Academy of Sciences*, 76(1):396, 1979.
- J.F. Crow, M. Kimura, et al. *An introduction to population genetics theory*. Harper & Row, Publishers, New York, Evanston, and London, 1970.
- T.S. Dahl, M.J. Matarić, and G.S. Sukhatme. Multi-robot task allocation through vacancy chain scheduling. *Robotics and Autonomous Systems*, 57(6-7):674–687, 2009.
- Ch. Darwin. *On the Origin of Species (1st edition)*, chapter 4, page 88. John Murray, 1859.
- K. De Jong. Learning with genetic algorithms: An overview. *Machine Learning*, 3(2-3): 121–138, 1988.
- K. A. De Jong. *Evolutionary Computation. A unified approach*. MIT Press, Cambridge, MA, USA, 2007.
- K.A. De Jong. Genetic algorithms are not function optimizers. In D. Whitley, editor, *Proceedings of the Second Workshop on Foundations of Genetic Algorithms*, pages 5–17, 1993.
- K.A. De Jong and W.M. Spears. A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of Mathematics and Artificial Intelligence*, 5(1):1–26, 1992.
- D.L. DeAngelis and W.M. Mooij. Individual-based modeling of ecological and evolutionary processes. *Annual Review of Ecology, Evolution, and Systematics*, 36(1):147–168, 2005.
- D. Doherty and C. O’Riordan. Evolving team behaviours in environments of varying difficulty. *Artificial Intelligence Review*, 27(4):223–244, 2007.
- M. Dorigo, V. Trianni, E. Şahin, R. Groß, T.H. Labella, G. Baldassarre, S. Nolfi, J.L. Deneubourg, F. Mondada, D. Floreano, et al. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2):223–245, 2004.
- M. Dorigo, M. Birattari, and T. Stutzle. Ant colony optimization. *IEEE Computational Intelligence Magazine*, 1(4):28–39, 2006.

- M. Dorigo, D. Floreano, LM Gambardella, F. Mondada, S. Nolfi, T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, et al. Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics & Automation Magazine*, in press.
- A Dornhaus. Specialization does not predict individual efficiency in an ant. *PLoS Biol*, 6 (11):e285, 2008.
- A. Dornhaus and NR Franks. Colony size affects collective decision-making in the ant *temnothorax albipennis*. *Insectes sociaux*, 53(4):420–427, 2006.
- A Dornhaus, JA Holley, and NR Franks. Larger colonies do not have more specialized workers in the ant *Temnothorax albipennis*. *Behav Ecol*, 20(5):922, 2009.
- A. Dornhaus, S. Powell, and S. Bengston. Group size and its effects on collective organization. *Annual review of entomology*, 57:123–141, 2012.
- A. Duarte, FJ Weissing, I Pen, and L Keller. An evolutionary perspective on self-organized division of labor in social insects. *Annual Review of Ecology, Evolution, and Systematics*, 42(1):91–110, 2011.
- A. Duarte, E. Scholtens, and F.J. Weissing. Implications of behavioral architecture for the evolution of self-organized division of labor. *PLoS Comp Biol*, 8(3):e1002430, 2012.
- R.C. Eberhart, Y. Shi, and J. Kennedy. *Swarm intelligence*. Morgan Kaufmann, 2001.
- A. Eiben, G. Nitschke, and M. Schut. Collective specialization for evolutionary design of a multi-robot system. In *Swarm Robotics*, volume 4433 of *Lecture Notes in Computer Science*, pages 189–205. Springer, 2007.
- J.A. Endler. *Natural selection in the wild*, volume 21. Princeton University Press, Princeton, NJ, USA, 1986.
- I. Eshel and L.L. Cavalli-Sforza. Assortment of encounters and evolution of cooperativeness. *Proceedings of the National Academy of Sciences*, 79(4):1331, 1982.
- L.J. Eshelman, R.A. Caruana, and J.D. Schaffer. Biases in the crossover landscape. In J.D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms (ICGA 1989)*, pages 10–19. Morgan Kaufmann Publishers Inc., 1989.
- C. Ferreira. Gene expression programming: A new adaptive algorithm for solving problems. *Complex Systems*, 13(2):87–129, 2001.

Bibliography

- R. Ferriere and R.E. Michod. Inclusive fitness in evolution. *Nature*, 471(7339):E6–E8, 2011.
- J.H. Fewell. Social insect networks. *Science*, 301(5641):1867, 2003.
- J.H. Fewell and R.E. Page Jr. Colony-level selection effects on individual and colony foraging task performance in honeybees, *Apis mellifera* L. *Behavioral Ecology and Sociobiology*, 48(3):173–181, 2000.
- S.G. Ficici, R.A. Watson, and J.B. Pollack. Embodied evolution: A response to challenges in evolutionary robotics. In J. Wyatt and Demiris J., editors, *Proceedings of the 8th European Workshop on Learning Robots*, volume 1812 of *Lecture Notes in Artificial Intelligence*, pages 14–22. Springer, 1999.
- S.G. Ficici, O. Melnik, and J.B. Pollack. A game-theoretic and dynamical-systems analysis of selection methods in coevolution. *IEEE Transactions on Evolutionary Computation*, 9(6):580–602, 2005.
- D. Floreano and L. Keller. Evolution of adaptive behaviour in robots by means of darwinian selection. *PLoS biology*, 8(1):e1000292, 2010.
- D. Floreano and J. Urzelai. Neural morphogenesis, synaptic plasticity, and evolution. *Theory in Biosciences*, 120(3):225–240, 2001.
- D. Floreano, S. Mitri, S. Magnenat, and L. Keller. Evolutionary conditions for the emergence of communication in robots. *Current biology*, 17(6):514–519, 2007.
- D. Floreano, P. Dürri, and C. Mattiussi. Neuroevolution: from architectures to learning. *Evolutionary Intelligence*, 1(1):47–62, 2008.
- D.B. Fogel. Evolving behaviors in the iterated prisoner’s dilemma. *Evolutionary Computation*, 1(1):77–97, 1993.
- D.B. Fogel. An introduction to simulated evolutionary optimization. *IEEE Transactions on Neural Networks*, 5(1):3–14, 1994.
- L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley and Sons, New York, 1966.
- J.A. Foster. Evolutionary computation. *Nature Reviews Genetics*, 2(6):428–436, 2001.
- K.R. Foster, T. Wenseleers, and F.L.W. Ratnieks. Kin selection is the key to altruism. *Trends in Ecology & Evolution*, 21(2):57–60, 2006.

- J.H. Fowler. Altruistic punishment and the origin of cooperation. *Proceedings of the National Academy of Sciences of the United States of America*, 102(19):7047, 2005.
- M. Franceschelli, A. Giua, and C. Seatzu. A gossip-based algorithm for discrete consensus over heterogeneous networks. *Automatic Control, IEEE Transactions on*, 55(5):1244–1249, 2010.
- N.R. Franks and C. Tofts. Foraging for work: how tasks allocate workers. *Animal Behaviour*, 48:470–472, 1994.
- T. Freeman, K. Keahey, I. Foster, A. Rana, B. Sotomoayor, and F. Würthwein. Division of labor: Tools for growing and scaling grids. *Service-Oriented Computing–ICSOC 2006*, pages 40–51, 2006.
- M. Genovart, N. Negre, G. Tavecchia, A. Bistuer, L. Parpal, and D. Oro. The young, the weak and the sick: evidence of natural selection by predation. *PloS one*, 5(3):e9774, 2010.
- B.P. Gerkey and M.J. Matarić. Sold!: Auction methods for multirobot coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.
- B.P. Gerkey and M.J. Matarić. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9):939–954, 2004.
- J.H. Gillespie. Natural selection for within-generation variance in offspring number ii. discrete haploid models. *Genetics*, 81(2):403–413, 1975.
- J.H. Gillespie. Natural selection for variances in offspring numbers: a new evolutionary principle. *The American Naturalist*, 111(981):1010–1014, 1977.
- P. Godfrey-Smith and B. Kerr. Selection in ephemeral networks. *The American Naturalist*, 174(6):906–911, 2009.
- S. Goings, J. Clune, C. Ofria, and R.T. Pennock. Kin selection: The rise and fall of kin-cheaters. In J. Pollack, M. Bedau, P. Husbands, T. Ikegami, and R. Watson, editors, *Artificial Life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*, pages 303–308. The MIT Press, 2004.
- D. E. Goldberg. *Genetic algorithms in search, optimization and machine learning*. Addison Wesley, Boston, MA, USA, 1989.
- D.E. Goldberg and K. Deb. A comparative analysis of selection schemes used in genetic algorithms. In *Foundations of genetic algorithms*, pages 69–93. Morgan Kaufmann Publishers, 1991.

Bibliography

- H.J. Goldsby, A. Dornhaus, B. Kerr, and C. Ofria. Task-switching costs promote the evolution of division of labor and shifts in individuality. *Proceedings of the National Academy of Sciences*, 109(34):13686–13691, 2012.
- D.M. Gordon. Dynamics of task switching in harvester ants. *Animal Behaviour*, 38(2):194–204, 1989.
- D.M. Gordon. The organization of work in social insect colonies. *Nature*, 380(6570):121–124, 1996.
- D.M. Gordon. The organization of work in social insect colonies. *Complexity*, 8(1):43–46, 2003.
- D.M. Gordon. Control without hierarchy. *Nature*, 446(7132):143–143, 2007.
- D.M. Gordon, B.C. Goodwin, and LEH Trainor. A parallel distributed model of the behaviour of ant colonies. *Journal of theoretical Biology*, 156(3):293–307, 1992.
- D.M. Gordon, S. Holmes, and S. Nacu. The short-term regulation of foraging in harvester ants. *Behavioral Ecology*, 19(1):217–222, 2008.
- S. Graham, M.R. Myerscough, J.C. Jones, and B.P. Oldroyd. Modelling the role of intracolony genetic diversity on regulation of brood temperature in honey bee (*Apis mellifera* L.) colonies. *Insectes Sociaux*, 53:226–232, 2006.
- M.J. Greene and D.M. Gordon. Interaction rate informs harvester ant task decisions. *Behavioral Ecology*, 18(2):451–455, 2007a.
- M.J. Greene and D.M. Gordon. How patrollers set foraging direction in harvester ants. *The American Naturalist*, 170(6):943–948, 2007b.
- J.J. Grefenstette. Credit assignment in rule discovery systems based on genetic algorithms. *Machine Learning*, 3(2):225–245, 1988.
- A.S. Griffin, S.A. West, and A. Buckling. Cooperation and competition in pathogenic bacteria. *Nature*, 430(7003):1024–1027, 2004.
- W. D. Hamilton. Selection of selfish and altruistic behaviour in some extreme models. In *Man and Beast: Comparative Social Behavior*, pages 59–91. Smithsonian Institutions Press, 1971.
- W.D. Hamilton. The genetical evolution of social behaviour i+ii. *Journal of Theoretical Biology*, 7(1):1–52, 1964.

- W.D. Hamilton. Selfish and spiteful behaviour in an evolutionary model. *Nature*, 228 (5277):1218–1220, 1970.
- A. Hara and T. Nagao. Emergence of cooperative behavior using adg: Automatically defined groups. In W. Banzhaf, J.M. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M.J. Jakiela, and R.E. Smith, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 1999)*, pages 1038–1046. Morgan Kaufmann, 1999.
- C. Hauert and M. Doebeli. Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature*, 428(6983):643–646, 2004.
- S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall, Upper Saddle River, 2nd edn edition, 1998.
- T. Haynes and S. Sen. Crossover operators for evolving a team. In J.R. Koza, B. Deb, M. Dorigo, D.B. Fogel, M. Garzon, H. Iba, and R.L. Riolo, editors, *Proceedings of the Second Annual Conference on Genetic Programming*, pages 162–167. Morgan Kaufmann, 1997.
- T.D. Haynes and S. Sen. Co-adaptation in a team. *International Journal of Computational Intelligence and Organizations*, 1(4):1–20, 1996.
- A.E. Hirsh and D.M. Gordon. Distributed problem solving in social insects. *Annals of Mathematics and Artificial Intelligence*, 31(1):199–221, 2001.
- J. H. Holland. *Adaptation in natural and artificial systems*. MIT Press, Cambridge, MA, USA, 1992.
- B. Hölldobler and E.O. Wilson. *The ants*. Belknap Press, Cambridge, 1990.
- J. Horn and D.E. Goldberg. Genetic algorithm difficulty and the modality of fitness landscapes. In L.D. Whitley and Vose M.D., editors, *Proceedings of the Third Workshop on Foundations of Genetic Algorithms*, pages 243–269. Morgan Kaufmann, 1994.
- C. Hou, M. Kaspari, H.B. Vander Zanden, and J.F. Gillooly. Energetic basis of colonial living in social insects. *Proceedings of the National Academy of Sciences*, 107(8):3634–3638, 2010.
- Z.Y. Huang and G.E. Robinson. Honeybee colony integration: worker-worker interactions mediate hormonally regulated plasticity in division of labor. *Proceedings of the National Academy of Sciences*, 89(24):11726–11729, 1992.
- Z.Y. Huang and G.E. Robinson. Regulation of honey bee division of labor by colony age demography. *Behavioral Ecology and Sociobiology*, 39(3):147–158, 1996.

Bibliography

- B.A. Huberman and N.S. Glance. Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences*, 90(16):7716–7718, 1993.
- L.D. Hurst. Genetics and the understanding of selection. *Nature Reviews Genetics*, 10(2): 83–93, 2009.
- H. Iba. Emergent cooperation for multiple agents using genetic programming. In Voigt H.-M., W. Ebeling, I. Rechenberger, and H.-P. Schwefel, editors, *Proceedings of the 4th International Conference on Parallel Problem Solving from Nature*, pages 32–41. Springer-Verlag, 1996.
- K.K. Ingram, P. Oefner, and D.M. Gordon. Task-specific expression of the foraging gene in harvester ants. *Molecular Ecology*, 14(3):813–818, 2005.
- I. Ispolatov, M. Ackermann, and M. Doebeli. Division of labour and the evolution of multicellularity. *Proceedings of the Royal Society B: Biological Sciences*, 279(1734): 1768–1776, 2012.
- R Jeanson, JH Fewell, R Gorelick, and SM Bertram. Emergence of increased division of labor as a function of group size. *Behavioral Ecology and Sociobiology*, 62:289–298, 2007.
- B.R. Johnson. Organization of work in the honeybee: a compromise between division of labour and behavioural flexibility. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270(1511):147–152, 2003.
- B.R. Johnson. Spatial effects, sampling errors, and task specialization in the honey bee. *Insectes Sociaux*, 57(2):239–248, 2010.
- C. Jones and M.J. Matarić. Adaptive division of labor in large-scale minimalist multi-robot systems. In *Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 2, pages 1969–1974. IEEE, 2003.
- J.C. Jones, M.R. Myerscough, S. Graham, and B.P. Oldroyd. Honey bee nest thermoregulation: diversity promotes stability. *Science*, 305(5682):402–404, 2004.
- G.E. Julian and S. Cahan. Undertaking specialization in the desert leaf-cutter ant *acromyrmex versicolor*. *Animal Behaviour*, 58(2):437–442, 1999.
- N. Kalra and A. Martinoli. Comparative Study of Market-Based and Threshold-Based Task Allocation. In *Distributed Autonomous Robotic Systems 7*, pages 91–101. Springer, 2006.

- J.R. Karr, J.C. Sanghvi, D.N. Macklin, M.V. Gutschow, J.M. Jacobs, B. Bolival, N. Assad-Garcia, J.I. Glass, and M.W. Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.
- N. Kashtan, E. Noor, and U. Alon. Varying environments can speed up evolution. *Proceedings of the National Academy of Sciences*, 104(34):13711, 2007.
- N. Kohl and R. Miikkulainen. Evolving neural networks for strategic decision-making problems. *Neural Networks*, 22(3):326–337, 2009.
- J. R. Koza. *Genetic Programming*. MIT Press, Cambridge, MA, USA, 1992.
- J.R. Koza. Evolving the architecture of a multi-part program in genetic programming using architecture-altering operations. In J. R. McDonnell, R. G. Reynolds, and D. B. Fogel, editors, *Evolutionary Programming IV: Proceedings of the Fourth Annual Conference on Evolutionary Programming*, A Bradford Book: Complex Adaptive Systems series, pages 695–718. The MIT Press, 1995.
- M.J.B. Krieger, J.B. Billeter, and L. Keller. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406(6799):992–995, 2000.
- P Kryger, U Kryger, and RFA Moritz. Genotypical variability for the tasks of water collecting and scenting in a honey bee colony. *Ethology*, 106(9):769–779, 2000.
- T.H. Labella, M. Dorigo, and J.L. Deneubourg. Division of labor in a group of robots inspired by ants’ foraging behavior. *ACM Transactions on Autonomous and Adaptive Systems*, 1(1):4–25, 2006.
- M. Lagoudakis, E. Markakis, D. Kempe, P. Keskinocak, A. Kleywegt, S. Koenig, C. Tovey, A. Meyerson, and S. Jain. Auction-based multi-robot routing. In *Robotics: Science and Systems*, pages 343–350. MIT Press, 2005.
- E. Leadbeater and L. Chittka. Social learning in insects—from miniature brains to consensus building. *Current Biology*, 17(16):R703–R713, 2007.
- L. Lehmann and F. Balloux. Natural selection on fecundity variance in subdivided populations: kin selection meets bet hedging. *Genetics*, 176(1):361–377, 2007.
- L. Lehmann and L. Keller. The evolution of cooperation and altruism—a general framework and a classification of models. *Journal of evolutionary biology*, 19(5):1365–1376, 2006.

Bibliography

- O. Leimar and P. Hammerstein. Evolution of cooperation through indirect reciprocity. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 268(1468):745–753, 2001.
- S. Lek and J.F. Guégan. Artificial neural networks as a tool in ecological modelling, an introduction. *Ecological Modelling*, 120(2-3):65–73, 1999.
- P. Lichocki, K. Krawiec, and W. Jaśkowski. Evolving teams of cooperating agents for real-time strategy game. In M. Giacobini, A. Brabazon, S. Cagnoni, G.A. Di Caro, A. Ekárt, A.I. Esparcia-Alcázar, M. Farooq, A. Fink, and P. Machado, editors, *Proceedings of the EvoWorkshops 2009 on Applications of Evolutionary Computing*, volume 5484 of *Lecture Notes in Computer Science*, pages 333–342. Springer, 2009.
- P. Lichocki, D. Tarapore, L. Keller, and D. Floreano. Neural networks as mechanisms to regulate division of labor. *The American Naturalist*, 179(3):391–400, 2012.
- P. Lichocki, L. Keller, and D. Floreano. Evolving team composition by agent swapping. *IEEE Transactions on Evolutionary Computations*, early access, in press.
- E. Lieberman, C. Hauert, and M.A. Nowak. Evolutionary dynamics on graphs. *Nature*, 433(7023):312–316, 2005.
- Y. Liu, X. Yao, and T. Higuchi. Evolutionary ensembles with negative correlation learning. *IEEE Transactions on Evolutionary Computation*, 4(4):380–387, 2000.
- S. Luke. *A User Manual for the ECJ Evolutionary Computation Library*, 2010.
- S. Luke and L. Spector. Evolving teamwork and coordination with genetic programming. In J.R. Koza, D.E. Goldberg, D.B. Fogel, and R.L. Riolo, editors, *Proceedings of the First Annual Conference on Genetic Programming*, pages 150–156, Cambridge, MA, 1996. MIT Press.
- S. Luke, C. Hohn, J. Farris, G. Jackson, and J. Hendler. Co-evolving soccer softbot team coordination with genetic programming. In *RoboCup-97: Robot soccer world cup I*, pages 398–411. Springer, 1998.
- A.C. Mailleux, J.L. Deneubourg, and C. Detrain. How does colony growth influence communication in ants? *Insectes Sociaux*, 50(1):24–31, 2003.
- D.P. Mandic and J.A. Chambers. *Recurrent neural networks for prediction: learning algorithms, architectures and stability*. Wiley, Chichester, 2001.

- D. Marocco, A. Cangelosi, and S. Nolfi. The emergence of communication in evolutionary robots. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 361(1811):2397–2421, 2003.
- M.J. Matarić, G.S. Sukhatme, and E.H. Østergaard. Multi-robot task allocation in uncertain environments. *Autonomous Robots*, 14(2):255–263, 2003.
- G.M. Mathews, H. Durrant-Whyte, and M. Prokopenko. Decentralised decision making in heterogeneous teams using anonymous optimisation. *Robotics and Autonomous Systems*, 57(3):310–320, 2009.
- JL McClelland, DE Rumelhart, and the PDP Research Group. *Parallel distributed processing, Vol. 1: Foundations and Vol. 2: Psychological and Biological Models*. The MIT Press, Cambridge, 1986.
- N. Meuleau and C. Lattaud. The artificial evolution of cooperation. In *Artificial Evolution*, volume 1063 of *Lecture Notes on Computer Science*, pages 159–180. Springer, 1996.
- T. Miconi. A collective genetic algorithm. In L. Spector, E.D. Goodman, A. Wu, Langdon W.B., Voigt H.-M., M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon, and E. Burke, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, pages 876–883. Morgan Kaufmann, 2001.
- T. Miconi. When evolving populations is better than coevolving individuals: the blind mice problem. In G. Gottlob and T. Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 647–652, San Francisco, CA, 2003. Morgan Kaufmann Publishers Inc.
- R. Milkman. Selection differentials and selection coefficients. *Genetics*, 88(2):391, 1978.
- M. Mirolli and D. Parisi. How can we explain the emergence of a language that benefits the hearer but not the speaker? *Connection Science*, 17(3):307–324, 2005.
- M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA, 1996.
- S. Mitri, D. Floreano, and L. Keller. The evolution of information suppression in communicating robots with conflicting interests. *Proceedings of the National Academy of Sciences*, 106(37):15786–15790, 2009.
- S. Mitri, J.B. Xavier, and K.R. Foster. Social evolution in multispecies biofilms. *Proceedings of the National Academy of Sciences*, 108(Supplement 2):10839–10846, 2011.

Bibliography

- S. Mitri, S. Wischmann, D. Floreano, and L. Keller. Using robots to understand social behaviour. *Biological Reviews*, 88(1):31–39, 2013.
- T. Nakashima, M. Takatani, M. Udo, H. Ishibuchi, and M. Nii. Performance evaluation of an evolutionary method for robocup soccer strategies. In A. Bredendfeld, A. Jacoff, I. Noda, and Y. Takahashi, editors, *RoboCup 2005: Robot Soccer World Cup IX*, Lecture Notes in Artificial Intelligence, pages 616–623. Springer, 2006.
- Y. Niv, D. Joel, I. Meilijson, and E. Ruppín. Evolution of reinforcement learning in uncertain environments: A simple explanation for complex foraging behaviors. *Adaptive Behavior*, 10(1):5–24, 2002.
- S. Nolfi and D. Floreano. Coevolving predator and prey robots: Do “arms races” arise in artificial evolution? *Artificial Life*, 4(4):311–335, 1998.
- S. Nolfi and D. Floreano. *Evolutionary Robotics: The Biology, Intelligence, and Technology*. MIT Press, Cambridge, MA, 2000.
- S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Teamwork in self-organized robot colonies. *Evolutionary Computation, IEEE Transactions on*, 13(4):695–711, 2009.
- M.A. Nowak. *Evolutionary dynamics: exploring the equations of life*. Belknap Press, 2006a.
- M.A. Nowak. Five rules for the evolution of cooperation. *Science*, 314(5805):1560–1563, 2006b.
- S. O’Donnell and R.L. Jeanne. Implications of senescence patterns for the evolution of age polyethism in eusocial insects. *Behavioral Ecology*, 6(3):269–273, 1995.
- B.P. Oldroyd, T.E. Rinderer, S.M. Buco, and L.D. Beaman. Genetic variance in honey bees for preferred foraging distance. *Animal Behaviour*, 45:323–332, 1993.
- R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.
- G.F. Oster and E.O. Wilson. *Caste and ecology in the social insects*. Princeton University Press, Princeton, 1978.
- E. Özcan, B. Bilgin, and E.E. Korkmaz. A comprehensive analysis of hyper-heuristics. *Intelligent Data Analysis*, 12(1):3–23, 2008.
- S.W. Pacala, D.M. Gordon, and H.C.J. Godfray. Effects of social group size on information transfer and task allocation. *Evolutionary Ecology*, 10(2):127–165, 1996.

- R.E. Page Jr and S.D Mitchell. Self-organization and the evolution of division of labor. *Apidologie*, 29(1-2):171–190, 1998.
- L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, 2005.
- L. Panait, S. Luke, and R.P. Wiegand. Biasing coevolutionary search for optimal multi-agent behaviors. *IEEE Transactions on Evolutionary Computation*, 10(6):629–645, 2006.
- C.A.C. Parker and H. Zhang. Cooperative decision-making in decentralized multiple-robot systems: The best-of-n problem. *Mechatronics, IEEE/ASME Transactions on*, 14(2):240–251, 2009.
- G.A. Parker and J.M. Smith. Optimality theory in evolutionary biology. *Nature*, 348(6296):27–33, 1990.
- L.E. Parker. Designing control laws for cooperative agent teams. In *Proceedings of 1993 IEEE International Conference on Robotics and Automation*, pages 582–587. IEEE, 1993.
- A. Pérez-Escudero, M. Rivera-Alba, and G.G. de Polavieja. Structure of deviations from optimality in biological systems. *Proceedings of the National Academy of Sciences*, 106(48):20544–20549, 2009.
- M. Potter and K. De Jong. A cooperative coevolutionary approach to function optimization. In Y. Davidor, H.-P. Schwefel, and R. Männer, editors, *Proceedings of the Third International Conference on Parallel Problem Solving from Nature (PPSN III)*, volume 866 of *Lecture Notes in Computer Science*, pages 249–257. Springer, 1994.
- M. A. Potter and D. V. De Jong. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation*, 8(1):1–29, 2000.
- S.C. Pratt. Quorum sensing by encounter rates in the ant *temnothorax albipennis*. *Behavioral Ecology*, 16(2):488–496, 2005.
- M. Preuss, N. Beume, H. Danielsiek, T. Hein, B. Naujoks, N. Piatkowski, R. Stüer, A. Thom, and S. Wessing. Towards intelligent team composition and maneuvering in real-time strategy games. *IEEE Transactions on Computational Intelligence and AI in Games*, 2(2):82–98, 2010.
- D.C. Queller. Kinship, reciprocity and synergism in the evolution of social behaviour. *Nature*, 318(6044):366–367, 1985.
- D.C. Queller. A general model for kin selection. *Evolution*, 46(2):376–380, 1992.

Bibliography

- M. Quinn. Evolving communication without dedicated communication channels. In *Advances in Artificial Life*, volume 2159 of *Lecture Notes on Computer Science*, pages 357–366. Springer, 2001.
- M. Quinn, L. Smith, G. Mayley, and P. Husbands. Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 361(1811):2321–2343, 2003.
- D.G. Rand, A. Dreber, T. Ellingsen, D. Fudenberg, and M.A. Nowak. Positive interactions promote public cooperation. *Science*, 325(5945):1272–1275, 2009.
- F.L.W. Ratnieks and C. Anderson. Task partitioning in insect societies. *Insectes Sociaux*, 46(2):95–108, 1999.
- F.L.W. Ratnieks, K.R. Foster, and T. Wenseleers. Conflict resolution in insect societies. *Annual Review of Entomology*, 51:581–608, 2006.
- F. Ravary, E. Lecoutey, G. Kaminski, N. Châline, and P. Jaisson. Individual experience alone can generate lasting division of labor in ants. *Current Biology*, 17(15):1308–1312, 2007.
- W. Ren, R.W. Beard, and E.M. Atkins. A survey of consensus problems in multi-agent coordination. In *American Control Conference, 2005. Proceedings of the 2005*, pages 1859–1864. IEEE, 2005.
- C.W. Reynolds. An evolved, vision-based behavioral model of coordinated group motion. In J.A. Meyer, H.L. Roitblat, and S.W. Wilson, editors, *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 384–392. Cambridge, MA: MIT Press, 1993.
- S.H. Rice. *Evolutionary theory: mathematical and conceptual foundations*. Sinauers Associate, Inc., Sunderland, MA, USA, 2004.
- S.H. Rice. A stochastic version of the price equation reveals the interplay of deterministic and stochastic processes in evolution. *BMC Evolutionary Biology*, 8(1):262, 2008.
- T.O. Richardson, E.J.H. Robinson, K. Christensen, H.J. Jensen, N.R. Franks, and A.B. Sendova-Franks. Record dynamics in ants. *PLoS one*, 5(3):e9621, 2010.
- R.L. Riolo, M.D. Cohen, and R. Axelrod. Evolution of cooperation without reciprocity. *Nature*, 414(6862):441–443, 2001.

- E.J.H. Robinson, O. Feinerman, and N.R. Franks. Flexible task allocation and the organization of work in ants. *Proceedings of the Royal Society B: Biological Sciences*, 276 (1677):4373–4380, 2009.
- G.E. Robinson. Modulation of alarm pheromone perception in the honey bee: evidence for division of labor based on hormonally regulated response thresholds. *Journal of Comparative Physiology A*, 160(5):613–619, 1987a.
- G.E. Robinson. Regulation of honey bee age polytheism by juvenile hormone. *Behavioral Ecology and Sociobiology*, 20(5):329–338, 1987b.
- G.E. Robinson. Regulation of division of labor in insect societies. *Annual Review of Entomology*, 37(1):637–665, 1992.
- G.E. Robinson and R.E. Page. Genetic determination of guarding and undertaking in honey-bee colonies. *Nature*, 333(6171):356–358, 1988.
- G.E. Robinson, C.M. Grozinger, and C.W. Whitfield. Sociogenomics: social life in molecular terms. *Nature Reviews Genetics*, 6(4):257–270, 2005.
- R. Rodriguez-Munoz, A. Bretman, J. Slate, CA Walling, and T. Tregenza. Natural and sexual selection in a wild insect population. *Science*, 328(5983):1269–1272, 2010.
- C. Rueffler, J. Hermisson, and G.P. Wagner. Evolution of functional specialization and division of labor. *Proceedings of the National Academy of Sciences*, 109(6):E326–E335, 2012.
- J.L. Sachs, U.G. Mueller, T.P. Wilcox, and J.J. Bull. The evolution of cooperation. *The Quarterly Review of Biology*, 79(2):135–160, 2004.
- E. Şahin. Swarm robotics: From sources of inspiration to domains of application. In *Swarm Robotics*, volume 3342 of *Lecture Notes in Computer Science*, pages 10–20. Springer, 2005.
- R.J. Schafer, S. Holmes, and D.M. Gordon. Forager activation and food availability in harvester ants. *Animal Behaviour*, 71(4):815–822, 2006.
- D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm. *Evolutionary Computation*, 1(1):25–49, 1993.
- P. Schmid-Hempel. Reproductive competition and the evolution of work load in social insects. *The American Naturalist*, 135(6):501–526, 1990.

Bibliography

- P. Schmid-Hempel and T. Wolf. Foraging effort and life span of workers in a social insect. *The Journal of Animal Ecology*, 57(2):509–521, 1988.
- M. Schneider-Fontan and M.J. Matarić. Territorial multi-robot task division. *IEEE Transactions on Robotics and Automation*, 14(5):815–822, 1998.
- T.W. Schoener. The newest synthesis: understanding the interplay of evolutionary and ecological dynamics. *Science*, 331(6016):426, 2011.
- M. Sebag and M. Schoenauer. Mutation by imitation in boolean evolution strategies. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors, *Proceedings of the 4th Conference on Parallel Problem Solving from Nature (PPSN IV)*, volume 1141 of *Lectures Notes in Computer Science*, pages 356–365. Springer, 1996.
- T.D. Seeley. Social foraging in honey bees: how nectar foragers assess their colony’s nutritional status. *Behavioral Ecology and Sociobiology*, 24(3):181–199, 1989.
- T.D. Seeley. *The wisdom of the hive*. Harvard University Press, Cambridge, 1995.
- T.D. Seeley, P.K. Visscher, T. Schlegel, P.M. Hogan, N.R. Franks, and J.A.R. Marshall. Stop signals provide cross inhibition in collective decision-making by honeybee swarms. *Science*, 335(6064):108–111, 2012.
- A.B. Sendova-Franks and N.R. Franks. Spatial relationships within nests of the ant *leptothorax unifasciatus* (latr.) and their implications for the division of labour. *Animal Behaviour*, 50(1):121–136, 1995.
- Y. Shoham and K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, Cambridge, 2009.
- E.D.V. Simoes and D.A.C. Barone. Predation: An approach to improving the evolution of real robots with a distributed evolutionary controller. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*, volume 1, pages 664–669. IEEE, 2002.
- A. Smith. *An Inquiry into the Nature and Causes of the Wealth of Nations*. Methuen and Co., Ltd., London, fifth (1904) edition, 1776.
- C.R. Smith, K.E. Anderson, C.V. Tillberg, J. Gadau, and A.V. Suarez. Caste determination in a polymorphic social insect: nutritional, social, and genetic factors. *The American Naturalist*, 172(4):497–507, 2008a.
- C.R. Smith, A.L. Toth, A.V. Suarez, and G.E. Robinson. Genetic and genomic analyses of the division of labour in insect societies. *Nature Reviews Genetics*, 9(10):735–748, 2008b.

- J.M. Smith. Byte-sized evolution. *Nature*, 355(6363):772–773, 1992.
- J.M. Smith and GR Price. The logic of animal conflict. *Nature*, 246:15, 1973.
- R.V. Solé, O. Miramontes, and B.C. Goodwin. Oscillations and chaos in ant societies. *Journal of theoretical Biology*, 161(3):343–357, 1993.
- T. Soule and R.B. Heckendorn. Improving performance and cooperation in multi-agent systems. In R. Riolo, T. Soule, and B. Worzel, editors, *Genetic Programming Theory and Practice V*, Genetic and Evolutionary Computation, pages 221–237. Springer, 2008.
- L. Spector, J. Klein, C. Perry, and M. Feinstein. Emergence of collective behavior in evolving populations of flying agents. *Genetic Programming and Evolvable Machines*, 6(1):111–125, 2005.
- A.J. Spencer, I.D. Couzin, and N.R. Franks. The dynamics of specialization and generalization within biological populations. *Advances in Complex Systems*, 1(01):115–127, 1998.
- K.O. Stanley, B.D. Bryant, and R. Miikkulainen. Real-time neuroevolution in the nero video game. *IEEE Transactions on Evolutionary Computation*, 9(6):653–668, 2005.
- P. Stone and M. Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383, 2000.
- Y. Suzuki and T. Arita. A comprehensive evaluation of the methods for evolving a cooperative team. *Artificial Life and Robotics*, 10(2):157–161, 2006.
- G. Syswerda. Uniform crossover in genetic algorithms. In J.D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms (ICGA 1989)*, pages 2–9. Morgan Kaufmann, 1989.
- D. Tarapore, D. Floreano, and L. Keller. Task-dependent influence of genetic architecture and mating frequency on division of labour in social insect societies. *Behavioral Ecology and Sociobiology*, 64(4):675–684, 2010.
- G. Theraulaz, E. Bonabeau, and JN Deneubourg. Response threshold reinforcements and division of labour in insect societies. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265(1393):327–332, 1998.
- M.L. Thomas and M.A. Elgar. Colony size affects division of labour in the ponerine ant *rhytidoponera metallica*. *Naturwissenschaften*, 90(2):88–92, 2003.

Bibliography

- C. Tofts. Algorithms for task allocation in ants.(a study of temporal polyethism: theory). *Bulletin of mathematical biology*, 55(5):891–918, 1993.
- H. Topcuoglu, S. Hariri, and M. Wu. Performance-effective and low-complexity task scheduling for heterogeneous computing. *Parallel and Distributed Systems, IEEE Transactions on*, 13(3):260–274, 2002.
- C. Tovey, M. Lagoudakis, S. Jain, and S. Koenig. The generation of bidding rules for auction-based robot coordination. *Multi-Robot Systems. From Swarms to Intelligent Automata Volume III*, pages 3–14, 2005.
- J.F.A. Traniello and R.B. Rosengaus. Ecology, evolution and division of labour in social insects. *Animal Behaviour*, 53(1):209–213, 1997.
- R.L. Trivers. The evolution of reciprocal altruism. *Quarterly review of biology*, 46(1):35–57, 1971.
- E. Tuci, M. Quinn, and I. Harvey. An evolutionary ecological approach to the study of learning behavior using a robot-based model. *Adaptive Behavior*, 10(3-4):201–221, 2002.
- G.P. Wagner. The measurement theory of fitness. *Evolution*, 64(5):1358–1376, 2010.
- G.P. Wagner and L. Altenberg. Perspective: Complex adaptations and the evolution of evolvability. *Evolution*, 50(3):967–976, 1996.
- M. Waibel, D. Floreano, S. Magnenat, and L. Keller. Division of labour and colony efficiency in social insects: effects of interactions between genetic architecture, colony kin structure and rate of perturbations. *Proceedings of the Royal Society B: Biological Sciences*, 273(1595):1815–1823, 2006.
- M. Waibel, L. Keller, and D. Floreano. Genetic team composition and level of selection in the evolution of cooperation. *IEEE Transactions on Evolutionary Computation*, 13(3): 648–660, 2009.
- M. Waibel, D. Floreano, and L. Keller. A quantitative test of hamilton’s rule for the evolution of altruism. *PLoS Biology*, 9(5):e1000615, 2011.
- JY Wakano, K. Nakata, and N. Yamamura. Dynamic model of optimal age polyethism in social insects under stable and fluctuating environments. *Journal of theoretical biology*, 193(1):153–165, 1998.
- C.R. Ward, F. Gobet, and G. Kendall. Evolving collective behavior in an artificial ecology. *Artificial Life*, 7(2):191–209, 2001.

- R.A. Watson, S.G. Ficici, and J.B. Pollack. Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1): 1–18, 2002.
- G.M. Werner and M.G. Dyer. Evolution of communication in artificial organisms. In C.G. Langton, C. Taylor, J.D. Farmer, and S. Rasmussen, editors, *Proceedings of the Second International Conference on Artificial Life*, volume X of *SFI Studies in the Sciences of Complexity*, pages 659–687. Addison-Wesley, 1991.
- S.A. West, A.S. Griffin, A. Gardner, and S.P. Diggle. Social evolution theory for microorganisms. *Nature Reviews Microbiology*, 4(8):597–607, 2006.
- T. White, B. Pagurek, and F. Oppacher. Asga: Improving the ant system by integration with genetic algorithms. In J.R. Koza, W. Banzhaf, K. Chellapilla, K. Deb, M. Dorigo, D.B. Fogel, M.H. Garzon, D.E. Goldberg, H. Iba, and R.L. Riolo, editors, *Proceedings of the Third Annual Conference on Genetic Programming*, pages 610–617, San Francisco, CA, 1998. Morgan Kaufmann.
- L.D. Whitley. Fundamental principles of deception in genetic search. In G.J.E. Rawlins, editor, *Proceedings of the First Workshop on Foundations of Genetic Algorithms*, pages 221–241. Morgan Kaufmann, 1991.
- R.P. Wiegand, W.C. Liles, and K.A. De Jong. An empirical analysis of collaboration methods in cooperative coevolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, pages 1235–1242, 2001.
- E.O. Wilson. *The insect societies*. Harvard University Press, Cambridge, MA, 1971.
- E.O. Wilson. Caste and division of labor in leaf-cutter ants (hymenoptera: Formicidae: Atta): 1. the overall pattern in *atta-sexdens*. *Behavioral Ecology and Sociobiology*, 7(2): 143–156, 1980.
- E.O. Wilson. Caste and division of labor in leaf-cutter ants (Hymenoptera: Formicidae: Atta): III. Ergonomic resiliency in foraging by *A. cephalotes*. *Behavioral Ecology and Sociobiology*, 14:47–54, 1983.
- E.O. Wilson. Between-caste aversion as a basis for division of labor in the ant *pleurogaster pubiventris* (hymenoptera: Formicidae). *Behavioral Ecology and Sociobiology*, 17(1): 35–37, 1985.
- E.O. Wilson and B. Hölldobler. Dense hierarchies and mass communication as the basis of organization in ant colonies. *Trends in Ecology & Evolution*, 3:65–68, 1988.

Bibliography

- S. Wischmann, D. Floreano, and L. Keller. Historical contingency affects signaling strategies and competitive abilities in evolving populations of simulated robots. *Proceedings of the National Academy of Sciences*, 109(3):864–868, 2012.
- W. Xiang and H.P. Lee. Ant colony intelligence in multi-agent dynamic manufacturing scheduling. *Engineering Applications of Artificial Intelligence*, 21(1):73–85, 2008.
- F. Zhang and C. Hui. Eco-evolutionary feedback and the invasion of cooperation in prisoner’s dilemma games. *PloS one*, 6(11):e27523, 2011.

CURRICULUM VITAE

General Information

Name and surname: Paweł Lichocki
E-mail: pawel.lichocki@gmail.com

Education

08.2009 – now École Polytechnique Fédérale de Lausanne, Ph.D. Program in Computer, Communication and Information Sciences
02.2007 – 09.2008 M.Sc., Poznań University of Technology, field of study: Computer Science, majoring in: Intelligent Decisions Support Systems
02.2008 – 06.2008 Exchange Student in the Erasmus Program at Paris Dauphine University (Paris-IX University), field of study: Computer Science
10.2003 – 02.2007 B.Sc. (engineer), Poznań University of Technology, field of study: Computer Science

Experience

08.2009 – now Doctoral Assistant, École Polytechnique Fédérale de Lausanne, Laboratory of Intelligent Systems
02.2012 – 05.2012 Software Engineer Intern, Google Zurich, EMEA Program
07.2008 – 07.2009 Analyst and Software Engineer, Institute of Bioorganic Chemistry – Poznań Supercomputing and Networking Center
10.2006 – 01.2008 Software Engineer, Institute of Bioorganic Chemistry – Poznań Supercomputing and Networking Center
10.2006 – 12.2006 Software Engineer and Product Manager, Poznań University of Technology – Institute of Computer Science
07.2006 – 09.2006 Scientist Visitor in Center for Computation and Technology at Louisiana State University
07.2005 – 12.2005 Software Engineer, Institute of Bioorganic Chemistry – Poznań Supercomputing and Networking Center

Major scholarships

10.2007 – 06.2008 Scholarship of the Polish Ministry of Science and Academic Education for outstanding academic accomplishments
03.2007 – 06.2007 Scholarship “Polish Talents” funded by Polish Telecommunication Group Foundation
10.2006 – 06.2007 Scholarship of the Polish Ministry of Science and Academic Education for outstanding academic accomplishments

Associations

09.2012 – now Association for Computing Machinery
01.2007 – now Institute of Electrical and Electronics Engineers, Inc.
02.2004 – 06.2008 Student Association for Computing Science at Poznań University of Technology

Awards

- The Best Oral Presentation and the Best Poster Presentation awards in Collective Dynamics Track on the Thirteenth International Conference on the Simulation and Synthesis of Living Systems (Artificial Life 13), 19-22 July 2012, East Lansing, Michigan, USA
- Computer Society International Design Competition 2006 Honorable Mention Team Award for project “IntelliForest”, 2 July 2006, Washington D.C., USA
- 2nd Grid Plugtest Flowshop Contest – second place, 10-14 October 2005, Sophia Antipolis, France

Peer-reviewed publications

- P. Lichocki, S. Wischmann, L. Keller, D. Floreano. **Evolving team compositions by agent swapping.** *IEEE Transactions on Evolutionary Computation*, early access, 2012
- P. Lichocki, D. Tarapore, L. Keller and D. Floreano. **Limitations of response thresholds models of division of labor.** *Artificial Life 13: Proceedings of the Thirteenth International Conference on the Simulation and Synthesis of Living Systems*, 561-562, Massachusetts: The MIT Press, 2012
- P. Lichocki, L. Keller and D. Floreano. **Differences in the Concept of Fitness Between Artificial Evolution and Natural Selection.** *Artificial Life 13: Proceedings of the Thirteenth International Conference on the Simulation and Synthesis of Living Systems*, 530-531, Massachusetts: The MIT Press, 2012
- P. Lichocki, D. Tarapore, L. Keller, D. Floreano. **Neural networks as mechanisms to regulate division of labor.** *The American Naturalist* 179 (3), 391-400, 2012
- M. Błażewicz, M. Ciżnicki, P. Kopta, K. Kurowski, P. Lichocki. **Two-dimensional discrete wavelet transform on large images for hybrid computing architectures: GPU and CELL.** *Euro-Par 2011: Parallel Processing Workshops, Revised Selected Papers, Part I*, Lecture Notes in Computer Science 7155, 481-490, Heidelberg: Springer, 2012
- P. Lichocki, P. Kahn Jr., A. Billard. **The ethical landscape of robotics.** *IEEE Robotics and Automation Magazine* 18 (1), 30-50, 2011
- K. Krawiec and P. Lichocki. **Using Co-solvability to model and exploit synergetic effects in evolution.** *Parallel Problem Solving from Nature - PPSN IX*, Lecture Notes in Computer Science 492-501, 492-501, Heidelberg: Springer, 2010
- K. Krawiec and P. Lichocki. **Approximating geometric crossover in semantic space.** *Proceedings of the 11th Annual conference on Genetic and Evolutionary Computation*, 987-994, New York: Association for Computing Machinery, 2009
- P. Lichocki, K. Krawiec and W. Jaśkowski. **Evolving teams of cooperating agents for real-time strategy game.** *Proceedings of the EvoWorkshops 2009 on Applications of Evolutionary Computing*, Lecture Notes in Computer Science 5484, 333-342, Heidelberg: Springer-Verlag, 2009

Additional Information

- known languages
 - Polish – native
 - English – very good
 - French – good
- professional interests
 - evolutionary theory and computation
 - analysis and simulations of complex systems
 - parallel and distributed computing
- personal interests
 - juggling
 - Origami
 - tennis
 - drawing