ARTICLE IN PRESS
Robotics and Autonomous Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Costs and benefits of behavioral specialization

Arne Brutschy [a,*], Nam-Luc Tran [a], Nadir Baiboun [a,b], Marco Frison [a,c], Giovanni Pini [a], Andrea Roli [c,a], Marco Dorigo [a], Mauro Birattari [a,*]

[a] IRIDIA, CoDE, Université Libre de Bruxelles, Avenue Franklin Roosevelt 50, 1050 Brussels, Belgium
[b] ECAM, Institut Supérieur Industriel, Rue du Tir 14, 1060 Brussels, Belgium
[c] DEIS-Cesena, Alma Mater Studiorum Università di Bologna, Via Montalti 69, 47521 Cesena, Italy

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Specialization
Task allocation
Swarm robotics
Swarm intelligence
Self-organization
Division of labor

ABSTRACT

In this work, we study behavioral specialization in a swarm of autonomous robots. In the studied swarm, robots have to carry out tasks of different types that appear stochastically in time and space in a given environment. We consider a setting in which a robot working repeatedly on tasks of the same type improves its performance on them due to learning. Robots can exploit learning by adapting their task selection behavior, that is, by selecting with higher probability tasks of the type on which they have improved their performance. This adaptation of behavior is called behavioral specialization. We employ a simple task allocation strategy that allows a swarm of robots to behaviorally specialize. We study the influence of different environmental parameters on the performance of the swarm and show that the swarm can exploit learning successfully. However, there is a trade-off between the benefits and the costs of specialization. We study this trade-off in multiple experiments using different swarm sizes. Our experimental results indicate that spatiality has a major influence on the costs and benefits of specialization.

1. Introduction

Division of labor is a concept that is common in the organization of large groups of individuals such as humans or social insects [1,2]. In division of labor, as defined for social insects by Beshers and Fewell, "(a) *each worker specializes in a subset of the complete repertoire of task types performed by the colony, and* (b) *this subset varies across individual workers in the colony*" [1]. In artificial systems, a common way to obtain division of labor is to let individuals adapt their behavior so that they predominantly work on a subset of the available task types—this is called *behavioral specialization* [3]. Behavioral specialization is known to increase the overall performance of an individual because of different reasons, one of them being learning. In some types of learning, an individual can acquire experience by repeatedly performing a task, which may improve the efficiency of the individual for tasks of the same type [4]. The individual can exploit this increased efficiency by adapting its task selection behavior, that is, by selecting with higher probability tasks of the type for which it has improved its performance.

In this work, which is an extension of [5], we study the costs and benefits of behavioral specialization in a swarm of autonomous robots. The robots of the swarm must perform two types of tasks. The tasks appear stochastically in time and in space and have to be carried out by the robots at the location where they appear. The spatial and temporal distributions of the tasks and their types are unknown to the robots. The robots can improve their performance on a certain type of task by repeatedly working on tasks of the same type (i.e., they learn). Learning is implemented using a simplified model that captures the relevant aspect of learning: robots get better in task performance upon repetition. This simplified model allows us to draw general conclusions on behavioral specialization without implementing a specific learning technique.

The robots of the swarm should adapt their behavior to work predominantly on a single type of task for two reasons: first, to learn to perform it more efficiently, and second, to fully exploit the resulting performance improvement. However, even though the exploitation of the performance improvement achieved through learning is clearly a benefit, specialization also entails certain costs [4]. An example of such costs is the time a specialized robot spends searching for a suitable task. For example, a fully specialized robot will work predominantly on tasks of a single type. An unspecialized robot, on the other hand, can work on the first task it encounters. In this example, depending on the amount of tasks of each type that are available in the environment, the specialized robot might spend more time searching for a suitable

* Corresponding authors.
E-mail addresses: arne.brutschy@ulb.ac.be (A. Brutschy), mbiro@ulb.ac.be (M. Birattari).

0921-8890/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2012.05.018

task than the unspecialized robot. Consequently, specialized robots can be less efficient than unspecialized robots when the tasks of the type in which they are specialized appear rarely. Therefore, specialization might be less advantageous in environments where the amount of tasks of each type and their spatial distribution frequently change.

In this work, robots face a task allocation problem: a robot must decide whether to engage in a task that it encounters in the environment. We employ two different strategies for task allocation. Robots using the *selective strategy* behaviorally specialize by selectively working on a specific type of task, thereby exploiting the performance improvement achievable through learning. Robots using the *greedy strategy* work on any task they encounter. We study the trade-off between costs and benefits of specialization by comparing the performance of the two strategies.

This article is organized as follows. In Section 2, we review related work. In Section 3, we describe how we model learning in the studied system. In Section 4, we describe the two different task allocation strategies that we use to study the costs and benefits of specialization. In Section 5, we describe the experimental setup that we use in our study. In Section 6, we describe the experiments and we report and discuss the results. In Section 7, we summarize the contributions of this work and present some directions for future research.

2. Related work

Specialization can be observed in many animals [6,7]. Especially animals that live in large groups, such as social insects, depend on specialization to efficiently organize the individuals of the group [1,6]. Most works that study specialization in social insects focus almost exclusively on specialization as a means of increasing task performance by reducing costs that are not directly related to the task execution itself, such as traveling between tasks [4]. In most of these works, individuals repeatedly perform a subset of tasks without improving in the actual execution of the tasks [7].

Specialization is widely studied using models that do not model embodiment, that is, the interactions between the physical body of the individuals and the environment. Such models are commonly used when studying the behavior of insect colonies [7,4]. Diwold et al. [8] studied the effect of the spatial organization of tasks on specialization by using agent-based models. In their work, they use *reinforced response thresholds* to attain specialization of individuals. The authors found that the studied systems achieve best performance when tasks are spatially separated. Note that the simplified model employed by the authors neither models spatial relationships between agents nor the cost of movement, that is, the model does not consider either losses of performance due to interference among multiple robots or losses due to time spent for traveling between tasks. Recently, Richardson et al. [9] studied a similar, threshold-based system that models spatiality using a statistical mechanics approach. The authors report that the more unequally labor is distributed among individuals, the higher is the resilience of the colony to external shocks.

A way of improving task performance, other than by reducing costs not directly related to the task execution itself, is learning. Higher vertebrates are known to exploit this type of improvement by behavioral specialization [10]. On the other hand, it is disputed if improvements of this kind can be observed in social insects [11]. In robotics, improving performance by learning is certainly possible, albeit complex to implement. See [12] for a survey of works on learning in multi-agent systems.

Even though some works exist that study specialization in terms of adapting behavior, few actually model or simulate spatiality and costs of specialization. Li et al. [13] studied division of labor and specialization in an initially homogeneous swarm, using a microscopic model that represents robots as separate probabilistic finite-state machines. In their work, robots behaviorally differentiate by assuming different roles in a stick-pulling experiment. The study confirms the observation that, in social insect colonies, specialization usually does not occur if the number of tasks is larger than the number of individuals. Hsieh et al. [14] extended the work of Li et al. by studying the system using a macroscopic analytical model based on continuous-time differential equations. Their results show that specialization might not be advantageous if the task-related parameters of the problem, such as the number of tasks in the environment, are known. If, on the other hand, there exists some uncertainty about these parameters, specialization is advantageous. Jones and Matarić [15] studied a foraging problem in which each individual specializes in foraging for one of two possible food types. The study shows that after a transition period the ratio of individuals specialized on either of the two food types matches the ratio of the food types in the environment. Murciano et al. [16] studied a system in which agents can specialize in foraging for one of two types of objects by learning an *affinity* for these types. The work uses reinforcement learning for adapting the behavior of the robots so that the ratio of robots specialized in foraging for either type of object matches the ratio of object types present in the environment.

A task frequently considered in studies on division of labor and specialization in a multi-robot system is foraging for energy. A common scenario is the study of two opposing behaviors: resting and foraging. The behaviors exhibit different costs and benefits in terms of energy: resting consumes little energy and does not yield energy, while foraging consumes large amounts of energy but can possibly yield energy by harvesting food items. The robots have to adapt their behavior in order to optimize their collective energy level. Labella et al. [17] found that, in their system, robots effectively divide into active and passive foragers. Liu et al. [18] studied a similar system that employs four different foraging strategies. Also this system exhibits an effective division of labor. Recently, Ikemoto et al. [19] proposed an adaptive mechanism for division of labor in a swarm of robots. The mechanism proposed divides the swarm into distinct groups that behaviorally specialize in a certain task.

3. Model of tasks, learning, and forgetting

In this section, we describe the type of tasks that must be executed by the robots of the swarm, and the effect that learning has on the performance of the robots.

We consider an instance of the single task/single robot task allocation problem [20]. That is, a task is carried out by a single robot, and a robot can work on a single task at a time. Also, robots can carry out tasks independently of each other.

The experimental environment that we consider consists of an arena that can be explored by the robots. A certain number of tasks are situated in specific locations within the arena. Robots can carry out a task when they are at the location of the task. Tasks appear stochastically with spatial and temporal distributions that are unknown to the robots. The goal of the swarm is to maximize its performance, measured as the number of tasks completed in a given period of time.

More specifically, we consider an environment in which robots can choose between two tasks: blue tasks and green tasks, denoted by τ_x with $x \in \{b, g\}$. To carry out a task, a robot has to reach the location of the task and stay there for a given amount of time, after which the task is completed.

While carrying out tasks, robots learn. To implement *learning*, we use a simple model that captures its most relevant effect on the performance of the robots: a robot that repeatedly performs a task of a certain type will become more efficient in performing other tasks of the same type. This simple model allows us to draw general conclusions on behavioral specialization without implementing any specific learning technique.

We also implement a form of *forgetting*: if a robot has improved its performance on a given task, and then either starts to work on another type of task or does not work on any task for some time, it loses part of its performance improvement for the first task type.

It is important to note that even though we refer to learning and forgetting with terms borrowed from studies of memory in humans or other animals, learning and forgetting can result from other processes. Examples are the morphological adaptation caused by muscle growth and loss (hypertrophy and atrophy) or the acquisition or loss of specialized tools. We keep our model of learning and forgetting deliberately coarse so that it can be used to describe any process that can cause improvement or degradation of task performance.

3.1. Learning

In our system, each time a robot executes a task, the robot improves its performance on that type of task by a given amount. This amount is not constant. We model the improvement in task performance analogously to what can be observed in natural systems: it increases rapidly for the first repeated executions of tasks of the same type, and reaches a plateau with further repetitions [11]. The improvement in task performance consists of a reduction of the task completion time w_x, that is, the time it takes to complete a task of type τ_x:

$$w_x(n_x) = \begin{cases} \overline{w}_{std} & \text{if } n_x = 0 \\ \overline{w}_{std} - \dfrac{\overline{w}_{std}}{k\,(1 + e^{-n_x + c})} & \text{if } 0 < n_x \le n_{max} \end{cases} \quad (1)$$

with $x \in \{b, g\}$. The meaning and effect of the parameter k and the constant c will be explained in the following. The counter n_x is incremented on the completion of a task of type τ_x, while, at the same time, the opposing counter n_y for task type τ_y, with $y \neq x$, is decremented (this is a form of forgetting; it is explained in detail in Section 3.2). Both counters are limited to the interval $[0, n_{max}]$. For example, if a robot has exclusively worked on tasks of type τ_b its counters are $n_b = n_{max}$ and $n_g = 0$.

The standard task completion time is denoted with $\overline{w}_{std}$; it is the time a robot takes to perform a task τ_x, when its $n_x = 0$. Note that in this work we use the same $\overline{w}_{std}$ for both task types in order to reduce the number of parameters of the system.

The factor k is used to vary the maximal time gain attainable through learning. This gain of learning is reached after a robot has successively completed n_{max} tasks of the same type. For convenience, we refer to the resulting minimal task completion time attainable by a fully learned robot as $\overline{w}_{min} = w_x(n_{max})$. Note that the parameter k is independent of $\overline{w}_{std}$, for example, $k = 1.25$ always results in a maximal time gain of 80%.

The constant $c = n_{max}/2$ renders the function $w_x(n_x)$ point-symmetric on the median of the interval $[0, n_{max}]$, that is, a robot reaches 50% of the time gain attainable through learning after performing $n_x = n_{max}/2$ tasks of type τ_x.

Fig. 1 shows a graphical representation of the learning model.

In the rest of the article, we refer to a specific setting of n_b and n_g as the *learning state* of a robot. The learning state thus designates the state of learning of a robot with respect to the two task types. Furthermore, we refer to the state $n_x = n_{max}$ as the *maximal learning state* for tasks of type τ_x (i.e., the robot has learned the best task execution possible), and to the state $n_x = 0$ as the *unlearned state* for tasks of type τ_x (i.e., the robot has not learned anything for this task type). In this article, we assume that the robots of the swarm are homogeneous at the beginning of the experiment with $n_b = n_g = 0$, that is, all robots are initialized to the unlearned state.

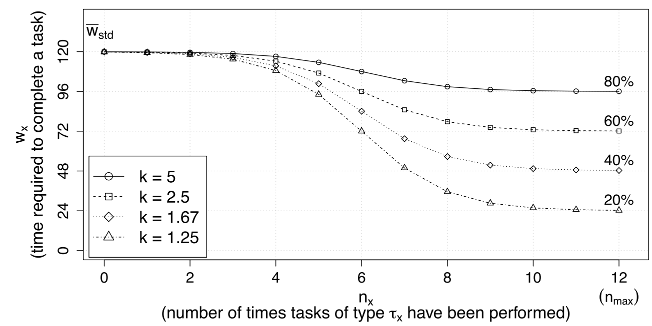


Fig. 1. The graph shows the effect of learning on the task completion time w_x for different values of the parameter k. Learning takes effect when a robot repeatedly works on the same type of task. Here, the standard task completion time in the unlearned state $\overline{w}_{std}$ is 120 s. The parameter k influences the time gain of learning. The values $k = \{1.25, 1.67, 2.5, 5\}$ shown here correspond to 20%, 40%, 60% and 80% of $\overline{w}_{std}$ at n_{max}, respectively.

3.2. Forgetting

In our learning model we also include the degradation of task performance, which we call *forgetting*. We implement forgetting as follows. First, when improving its performance on a certain type of task τ_x due to learning, a robot forgets what it learned previously about the other task type, that is, upon incrementing n_x, we decrement n_y, with $y \neq x$. Second, a robot that keeps searching for tasks of a certain type gradually decreases its performance for both task types, that is, upon traveling for a distance d_f, the counters n_b and n_g are both decremented by 1 (to a minimum of 0). This mechanism causes the robots to return to an unlearned state over time.

Forgetting can be observed in many natural systems, as it can improve the efficiency of individuals. Individuals have to spend energy in order to maintain adaptations, be it muscle- or memory-based. In case these adaptations are not advantageous in the current situation of the individual, it is beneficial for the individual to discard them to save energy. It has been shown that forgetting can also improve the performance of artificial systems such as robotic swarms [21].

4. Task allocation strategies

In this section, we explain the two strategies that are used by the robots of the swarm for allocating tasks. Robots using the *selective strategy* select among the tasks they encounter. This allows the robots to specialize in a certain type of task in order to exploit the performance improvement available through learning (see Section 3). Robots using the *greedy strategy* do not select among the tasks but work on any task they encounter.

4.1. Selective strategy

Robots using behavioral specialization adapt their behavior so that they work selectively on a certain type of task. Behavioral specialization therefore depends on the strategy used by the robots to allocate tasks. In this work, the robots employ a simple stochastic strategy to decide whether to accept a task they encounter in the environment. The strategy is fully distributed and requires no communication between robots, as it depends only on the robots' memory of the previously completed tasks. In the following, p_g denotes the probability that a robot accepts a task of type τ_g upon encountering it. The robot computes p_g as a function of the memory counter m of previously completed tasks:

$$p_g(m) = \frac{1}{1 + e^{-\gamma m}}, \quad (2)$$

with γ being a parameter that defines the steepness of the probability curve, referred to as task acceptance modifier. As the name indicates, γ influences the probability with which a robot accepts tasks of the same type: higher values of γ require a lower amount of tasks to be completed in order to reach the maximum probability, that is, the function $p_g(m)$ approximates a step function. The memory counter m of completed tasks is initialized to 0 at the beginning of the experiment. The robot increments m upon the execution of a task of type τ_g, and decrements it upon the execution of a task of type τ_b. The value of m is limited to the interval $[-10, 10]$; therefore, a robot has to perform 20 tasks of type τ_b to change from being fully specialized in τ_g ($m = 10$) to being fully specialized in τ_b ($m = -10$).

Note that, because in this article we only consider two different types of tasks, we can compute the probability of the robot to work on tasks of type τ_b as $p_b = 1 - p_g$. As a result, robots that worked repeatedly on one type of task adapt their behavior so that they reject with an increasing probability tasks of the other type. A robot reaching a state in which it works exclusively on a single task type τ_x is called a *specialist* for τ_x (i.e., $p_x \simeq 1$ and $p_y \simeq 0$, with $y \neq x$). Analogously, a robot that has no behavioral preference in task acceptance is called a *generalist* (i.e., $p_b = p_g \simeq 0.5$).

If a robot does not accept to work on a task it encounters, it continues searching for tasks by performing a random walk. In order to prevent dead-locks in the form of robots specialized in a type of task that is not available in the environment, robots forget their behavioral specialization while searching. This functionality is implemented by decreasing $|m|$ whenever the robot traveled for a distance $\hat{d}_f$. More specifically, m is decremented by 1 in case of $m > 0$ and incremented by 1 in case of $m < 0$. As a result, the probability p_x of a robot approaches 0.5 for both tasks, that is, the robot returns to a generalist behavior while searching for tasks. In order to reduce the number of parameters of the system, we set the forgetting distance $\hat{d}_f$ of the selective strategy to the forgetting distance d_f of the learning model.

The function described in Eq. (2) leads to the specialization of the robot to a specific task as follows. By performing a task of type τ_x, a robot increases its probability p_x to execute again tasks of this type. This causes the robot to repeatedly work on tasks of the same type (if available), thereby becoming a specialist for this type of task. Conversely, if a robot travels for a distance $\hat{d}_f$ without performing any task or if it performs tasks of the other type τ_y, with $y \neq x$, the probability of accepting tasks of type τ_x decreases.

4.2. Greedy strategy

Robots using the greedy strategy start to work on any task they encounter. Therefore, they do not behaviorally specialize on any of the task types and do not exploit the performance improvement available through learning. When a robots uses the greedy strategy, the probability to accept tasks is $p_b = p_g = 1$. Note however that, even though task allocation in the greedy strategy is random, robots can still improve their performance by learning; this is the case when a robot happens to work predominantly on tasks of the same type.

The greedy strategy provides a performance reference for comparison with the selective strategy. As robots using the greedy strategy do not specialize behaviorally, the direct comparison between the results of swarms using the two strategies allows us to evaluate the performance improvement available through behavioral specialization.

5. Experimental setup

In this section, we describe the setup of the experiments that we use to evaluate the costs and benefits of behavioral specialization. In the following, we describe the robots we use for the experiments, as well as the method used to represent the tasks that have to be performed by the robots. We assume the model of learning and forgetting presented in Section 3. Additionally, we present details about the simulation environment, the controller of the robots, and the arena employed in the experiments. Finally, we discuss the metrics we use for measuring diversity and specialization in the swarm.

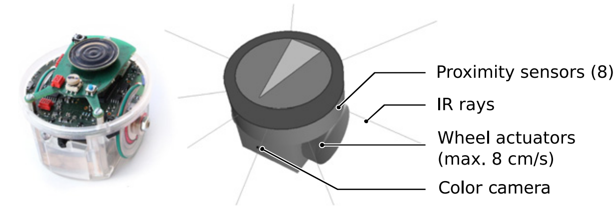


Fig. 2. The e-puck robot. Left: A picture of the real robot. Right: The e-puck as represented in simulation.

5.1. Robot

In the experiments presented in this paper, we simulate the commercial robot called *e-puck*.[1] The e-puck is a small wheeled robot, designed to be a research and educational tool for university students [22]. The e-puck features 8 infra-red proximity sensors that can also be used as light sensors, a forward-facing color camera (with a resolution of 640×480 pixels), 8 red LEDs and the wheel actuators. In the experiments presented in this article, we employ the wheel actuators (with a maximum speed of $s_{max} = 8$ cm/s), the proximity sensors for obstacle avoidance, and the camera for the detection of tasks. Note that the e-puck does not have any manipulation capabilities (Fig. 2).

5.2. Task abstraction

In order to overcome the lack of manipulation capabilities of the e-puck, we abstract the tasks that the robots can perform with a device called *task allocation module* (TAM) [23]. Fig. 3 illustrates the basic functionality of a TAM. Each TAM features a light barrier and two RGB LEDs. The LEDs can be perceived by a robot using its color camera. The robot can navigate to the TAM and enter it. The presence of a robot can be detected by the TAM using its light barrier. Upon the detection of a robot, the TAM reacts by changing the color of its LEDs following a user defined logic. TAMs can represent the interaction between the e-puck robot and the environment in many different settings, for example tasks that need to be executed by the robots [5] or material that needs to be transported [24].

In our experiments, each TAM represents one of the two types of tasks, τ_b and τ_g. The type of the task is encoded by the color of the LEDs of the TAM. Thus, a robot can perceive which type of task a TAM represents. If a robot enters a TAM, it is considered to work on the corresponding task. The TAM acknowledges the robot's presence by temporarily changing the color of its LEDs to red. The robot remains inside the TAM for the time w_x required to complete the task. This time is, in general, different from robot to robot, as it depends on the robot's learning state, regulated by the learning model explained in Section 3. After the robot has completed the task and has left, the TAM stochastically selects the type of the next task (i.e., the color). Thus, a TAM is always representing a task, that is, there are no idle TAMs.

[1] http://www.e-puck.org/.

Please cite this article in press as: A. Brutschy, et al., Costs and benefits of behavioral specialization, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.05.018

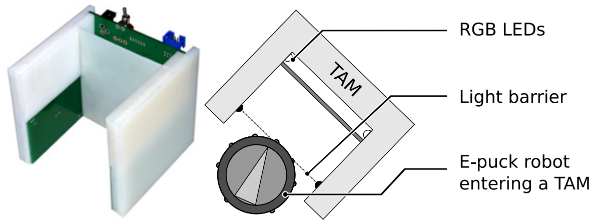


Fig. 3. A device, called *task allocation module* (TAM), used for abstracting the tasks that have to be performed by the robots. Left: A photo of the real device. Right: Functional schematics of a TAM. The light barrier detects a robot entering into the TAM. Upon the detection of a robot, the TAM reacts by changing the color of its LEDs following a user defined logic. In our experiments, the different task types are encoded by the LED color of the TAM.

We can change the distribution of task types in the environment by modifying the probability with which a TAM selects the type of a new task. Note that a change of this probability affects the distribution of the tasks only after a certain amount of time has elapsed, as a TAM only generates a new task after the previous task has been completed. We refer to the ratio of the blue tasks in the environment as *task ratio*, defined as $r = |\tau_b|/T$ with $T = |\tau_b| + |\tau_g|$ being the total number of TAMs, which equals the total number of tasks concurrently available in the environment.

5.3. Simulation tools

In the following we describe the tools we employ to simulate a swarm of e-puck robots and the tasks represented by the TAMs. More specifically, we describe the simulation framework used, the controller of the robots, and the arena.

5.3.1. Simulation framework

The work presented here has been carried out using the ARGoS simulation framework [25]. ARGoS is a discrete-time physics-based simulation framework developed within the Swarmanoid project [26].[2] ARGoS is open source and can be freely used for other research projects.[3] It can simulate various robots at different levels of detail. The experiments presented in this work are carried out in a 2-dimensional kinematics-based simulation. ARGoS simulates the whole set of sensors and actuators available on the e-puck. The TAM, including its sensors and actuators, is also simulated in ARGoS.

5.3.2. Robot controller

In our experiments, the two task types τ_b and τ_g are represented by the TAMs using blue or green LEDs, respectively. The robots can perceive tasks within a limited distance using their camera, and recognize their type by their color. Fig. 4 shows the behavior of the robots when using the selective strategy. A robot performs a random walk to search the environment for tasks that need to be executed. In case the robot encounters another robot or static obstacles, it performs an obstacle avoidance maneuver (the corresponding state has been omitted from Fig. 4). When a robot perceives a task τ_x, it applies its associated task allocation strategy as described in Section 4. In case the robot uses the selective strategy, it has an associated probability p_x to start to work on the perceived task (see Eq. (2)). In case the robot uses the greedy strategy, it starts to work on any task it perceives. Each robot is subject to the learning and forgetting mechanisms described in Section 3. The controller is behavior-based, that is, it is composed of several modules, each of which controls a distinct behavior of the robot. All the robots of the swarm use an instance of the same controller and start in an unlearned state (see Section 3).

[2] http://www.swarmanoid.org/.

[3] http://iridia.ulb.ac.be/argos/.

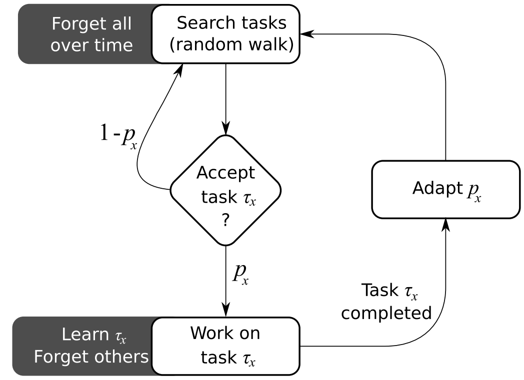


Fig. 4. Finite state machine describing the behavior of the robots when using the selective strategy. Light rectangles represent actions executed by the robot, dark rectangles show the effect of learning and forgetting on the robot. A robot accepts a task it encounters with probability p_x (see Eq. (2)).

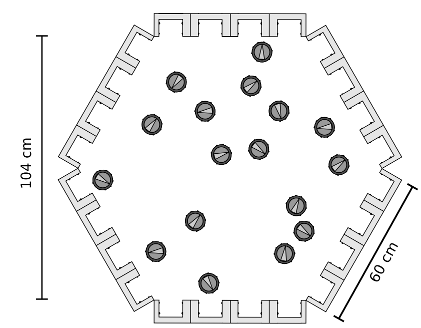


Fig. 5. Representation of the arena with e-pucks at random initial positions. Tasks are represented by the TAMs which are located at the boundaries of the arena, with a total of $T = 24$ tasks concurrently available in the environment. Each TAM stochastically selects which type of task it represents, that is, task types appear stochastically in time and space.

5.3.3. Environment

The environment consists of an obstacle free, hexagonal arena (see Fig. 5). The tasks to be performed are represented by TAMs at the boundaries of the arena. Each of the six sides of the arena consists of 4 TAMs, for a total of $T = 24$ tasks concurrently available in the environment. The distance between a TAM and the one facing it on the opposite side of the arena is 104cm. This distance is such that a robot that leaves a TAM cannot directly perceive the tasks of the TAM situated diametrically. This guarantees that a robot has to spend some time searching before encountering another task.

5.4. Measures of diversity and specialization

In the following we describe the measures we use for evaluating diversity and specialization in the swarm.

5.4.1. Diversity in the learning states

In the context of this article, we define the diversity of a swarm in terms of the diversification of the learning states among the robots of the swarm (see Section 3.1). This allows us to compare the diversity of a swarm using the selective strategy, that exploits the improvements available through learning, to the diversity of a swarm using the greedy strategy, that does not exploit the improvements available through learning. In order to measure the diversity in the swarm, we use the *hierarchic social entropy*, as initially proposed by Balch [27]. The hierarchic social entropy is based on hierarchical clustering of the robots and the *simple social entropy*, which in turn is based on Shannon's *information entropy* [28].

Please cite this article in press as: A. Brutschy, et al., Costs and benefits of behavioral specialization, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.05.018

The simple social entropy H measures the diversity of a swarm depending on a classification of the robots. Let $\mathcal{C}_h$ be a given classification of a swarm $\mathcal{R}$ of N robots into M possibly overlapping subsets c_i, with h being a parameter of the classification method, which is explained in the following. The simple social entropy H for the classification $\mathcal{C}_h$ can be computed as follows. Let the proportion of robots in the ith subset be $p_i = |c_i|/N$. We compute H (i.e., the Shannon index) of the classification $\mathcal{C}_h$ as

$$H(\mathcal{C}_h) = -\sum_{i=1}^{M} p_i \log p_i. \tag{3}$$

The value of the simple social entropy H depends on the classification $\mathcal{C}_h$ and therefore on the method used to derive this classification. We employ a method that classifies the robots on the basis of a distance, computed using a given distance metric. We consider two robots to belong to the same classification $\mathcal{C}_h$ if the distance between them is smaller than a threshold distance h.

In this article it is relevant to cluster robots according to their learning state as defined in Section 3.1. Therefore, the distance metric should reflect the diversity between two robots of the swarm. To this end, we define the distance metric d as the Euclidean distance between the learning state of two robots ρ_1 and ρ_2:

$$d(\rho_1, \rho_2) = \sqrt{(n_{b_1} - n_{b_2})^2 + (n_{g_1} - n_{g_2})^2}\,, \tag{4}$$

with n_b and n_g being, for both robots, the number of tasks, performed in the last n_{max} executions, of type τ_b and τ_g, respectively.

As mentioned above, $H(\mathcal{C}_h)$ is the simple social entropy for a classification of robots derived using the parameter h. In order to remove the dependency of the simple social entropy on this parameter, we can integrate over it and obtain the so-called hierarchic social entropy:

$$D(\mathcal{R}) = \int_0^\infty H(\mathcal{C}_h)\,\mathrm{d}h\,. \tag{5}$$

The value of the hierarchic social entropy $D(\mathcal{R})$ measures the diversity of a swarm, with higher values designating a higher diversity and $D(\mathcal{R}) = 0$ indicating a swarm that is completely homogeneous. See [27] for more detailed information about the hierarchic social entropy.

5.4.2. Specialization

As behavioral specialization in the swarm is not covered by the hierarchic social entropy, we require another measure in order to quantify the change in behavior of the robots. We define two measures that differ in the way they measure the behavior of the robots. The first one measures specialization in terms of transitions in the sequence of tasks performed by a robot, while the second one measures specialization in terms of the internal task acceptance probability of a robot. The former is referred to as *F measure*, and the latter is referred to as *P measure*.

The F measure is based on the frequency of switches between task types in the sequence of tasks completed by a robot. It has been developed by Gautrais et al. in their study of specialization in insect colonies [7]. The individual measure F_i is a value in the range $[-1, 1]$, representing the degree of specialization of a robot i. For a sequence of N_i tasks, it is computed as $F_i = 1 - (2S_i/N_i)$, where S_i is the number of times the robot i switched between task types. The value of F_i is 1 for a fully specialized robot, 0 for random task allocation, and -1 for systematic switching between task types. F is the average over the values of F_i of all robots of the swarm. Table 1 reports some examples of task sequences, the corresponding value of F, and the interpretation of this value. The F measure allows one to compare the behavior of the robots depending on the sequence of tasks performed. It is independent of the underlying mechanism used by the robots to select the type of the next task to tackle.

Table 1
Examples for the value of the F measure for different task sequences.

Task sequence	F	Interpretation
$\tau_g\tau_g\tau_g\tau_g\tau_g\tau_g\tau_g\tau_g$	1	Fully specialized robot
$\tau_g\tau_b\tau_g\tau_b\tau_g\tau_b\tau_g\tau_b$	~ -1	Systematic switching
$\tau_b\tau_b\tau_g\tau_b\tau_g\tau_g\tau_g\tau_b$	0	Random task allocation

The P measure is based on the internal probability of a robot to accept a task when encountering it. We consider a robot to be specialized in a task of type τ_x if its probability p_x of accepting a task of this type, as defined by Eq. (2), is greater or equal to 0.95. $P(\tau_x)$ is the number of robots specialized in task type τ_x according to this definition. As the P measure depends on the internal probability of a robot to accept a task, it is only applicable to the selective strategy.

Previous works have used measures of specialization different from the ones described above. Li et al. define specialization as *positive diversity*, that is, diversity that increases the performance of the swarm [13]. They base their specialization measure on the hierarchic social entropy by correlating it with the global performance of the swarm. This measure cannot cope with (a) the whole swarm specializing in the same task and (b) specialization leading to reduction in performance. O'Donnell and Jeanne also measure specialization using the Shannon index [28], but base it on the proportion of the tasks executed by an agent over the whole set of tasks available [29]. They define specialization as diversity in task acceptance, similarly to Li et al., but do not correlate diversity to performance. The measure cannot detect differences in the order of task acceptance as the one proposed by Gautrais et al. [7], that is, it cannot distinguish between the case of a robot switching constantly and a robot working exclusively first on one, and then on another task. The shortcomings of the measures used in [13,29] and their different focus led us to use the F measure proposed by Gautrais et al. in conjunction with the P measure for quantifying the behavioral specialization of the robots using the selective strategy.

6. Experiments

In this section, we describe the experiments that we performed to study the costs and benefits of behavioral specialization. Additionally, we present and discuss the results of these experiments.

For our experiments, we use the following parameter settings. For each experimental condition we conduct 20 randomly seeded runs, for a duration of $t_{max} = 10{,}000$ simulated seconds each. In general, we measure time t in seconds. The ratio r of blue tasks is 0.5 unless mentioned otherwise, that is, there is an equal probability of encountering one of the two types of tasks. The standard task completion time $\overline{w}_{std}$ is set to 120 s for both task types. The gain parameter of the learning model (Eq. (1)) is set to $k = 1.25$, with $\overline{w}_{min} = 24$ s. This results in a 80% gain in task completion time at the maximal learning state, which is reached after $n_{max} = 12$ consecutive executions of the same task type. The forgetting distance is $d_f = 300$ cm.

The swarm is composed of $N = 18$ e-pucks, randomly positioned in the arena at the beginning of each experimental run. The maximum speed s_{max} of the robots is set to 8 cm/s. The robots use the controller described in Section 5.3.2 and one of the two possible strategies described in Section 4. In case of the selective strategy, the task acceptance probabilities p_g and p_b are both initialized to 0.5, respectively; thus, all robots start in the unlearned state. The task acceptance modifier of Eq. (2) has been determined by trial

Table 2
Default parameters of the experiments.

Param.	Meaning	Value
N	Number of robots	18
T	Number of concurrent tasks	24
r	Task ratio	0.5
e	Number of experimental runs	20
t_{max}	Experiment duration	10,000 s
$\overline{w}_{std}$	Completion time if unlearned	120 s
$\overline{w}_{min}$	Completion time if fully learned	24 s
n_{max}	Tasks required to be fully learned	12
k	Learning curve parameter (gain)	1.25
d_f	Forgetting distance	300 cm
s_{max}	Maximum wheel speed	8 cm/s
$\hat{p}_g, \hat{p}_b$	Initial task acceptance probabilities	0.5
m	Memory counter of completed tasks	[−10, 10]
γ	Task acceptance modifier	1

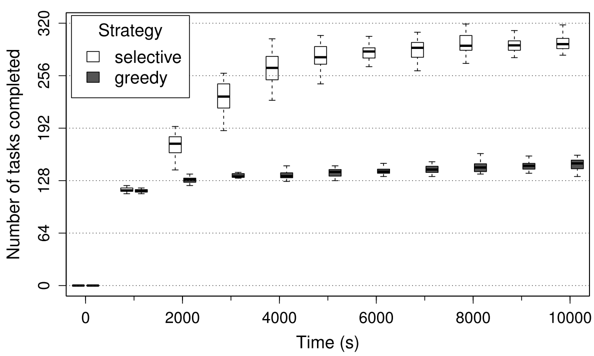


Fig. 6. Performance, defined as the total number of tasks completed in the preceding 1000 s, shown for each strategy over time. Performance is measured every 1000 s. Data collected over 20 experimental runs.

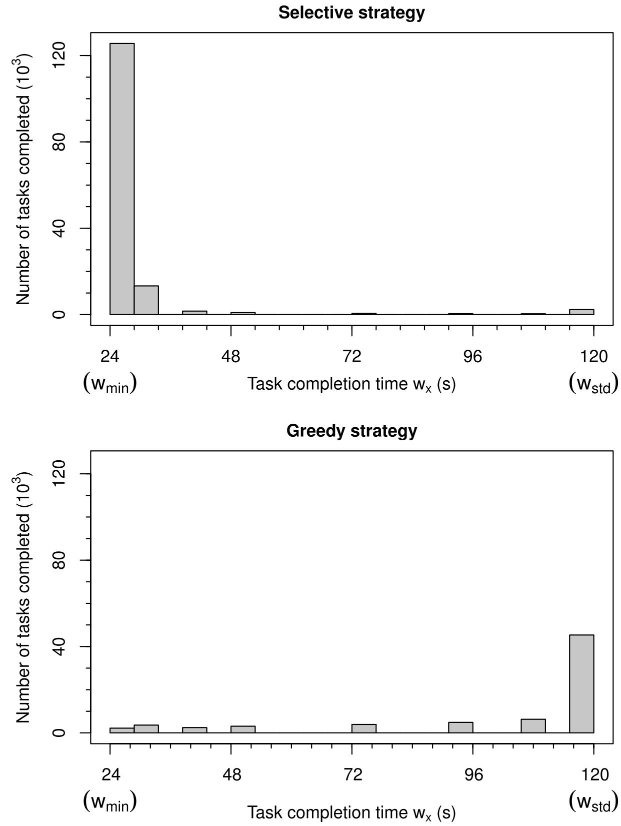


Fig. 7. Histograms reporting, for both strategies, the number of times robots have completed a task in w_x seconds, collected at the steady state (5000 s $\leq t \leq$ 10,000 s). Data collected over 20 experimental runs. Top: almost all robots using the selective strategy complete their task at the minimal task completion time, $\overline{w}_{min} = 24$. Bottom: robots using the greedy strategy mostly complete tasks at high task completion times.

and error in preliminary experiments and is set to $\gamma = 1$ (see on-line supplementary material [30]).

In general, we give a non-parametric description of data by reporting median and interquartile range (IQR) in the plots. In case we give numerical results in the text, we report the quartiles of the distribution in the format 25%/50%/75%. In case the observations are normally distributed, we provide a parametric description by reporting mean and standard deviation. In this case, and when it is necessary to determine if the difference between two values is statistically significant, we additionally report the results of a Welch's t-test. Table 2 summarizes all parameters of the environment, the learning model, and the robots.

6.1. Static environment

In the first set of experiments we employ a static environment, that is, the task ratio remains constant throughout this set. We use this set to study the basic properties of the system: How do the two strategies perform, and does specialization occur?

To answer these questions, we first assess if the selective task allocation strategy presented in Section 4.1 successfully exploits the performance improvement available through learning. To do so, we compare the performance of the selective strategy to the performance of the greedy strategy. Performance is defined as the amount of tasks completed in the preceding 1000 s; and we measure it every 1000 s. Fig. 6 shows the evolution of the performance of each strategy over time. In the plot, we report the median of the observation and the interquartile range (IQR). As it can be seen, after an initial period in which the swarm specializes, the selective strategy performs better than the greedy strategy. The observation of the performance for a given value of t is distributed normally (see supplementary on-line material [30]); in the following we therefore report the mean(std). The performance at the end of the experiment is 296.3(10.0) and 146.9(7.8), in case of the selective strategy and of the greedy strategy, respectively. The performance of the two strategies is significantly different starting from $t = 2000$ (Welch's t-test with $\alpha = 0.05$).

Next, we study to which degree the robots learn to work on a certain type of task. Fig. 7 reports, for both strategies, the number of times robots have completed a task in w_x seconds, collected at the steady state (5000 s $\leq t \leq$ 10,000 s). As the time w_x robots spend working on a task decreases with learning, a high amount of low task completion times indicates a high degree of learning in the swarm. Fig. 7 (top) shows that for robots using the selective strategy, most tasks have been completed at the minimal task completion time, $\overline{w}_{min} = 24$. This indicates that most robots work at the maximal learning state for one of the two task type, that is, the robots exploit learning to the full extent. Fig. 7 (bottom) shows that robots using the greedy strategy mostly complete tasks at high task completion times. This indicates that there is poor or no learning in the swarm. As task allocation is random when using the greedy strategy, some robots happen to improve their performance temporarily by repeatedly working on the same task type, thereby completing their tasks in shorter time. Nevertheless, as this behavior is not systematic, the greedy strategy cannot exploit the advantages offered by learning in a consistent manner.

Additionally, we study to which degree the robots behaviorally specialize in one of the two tasks. Fig. 8 shows a scatter plot of the total number of completed tasks per type for each robot during the course of the experiment. We also report the value of the hierarchic

Please cite this article in press as: A. Brutschy, et al., Costs and benefits of behavioral specialization, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.05.018

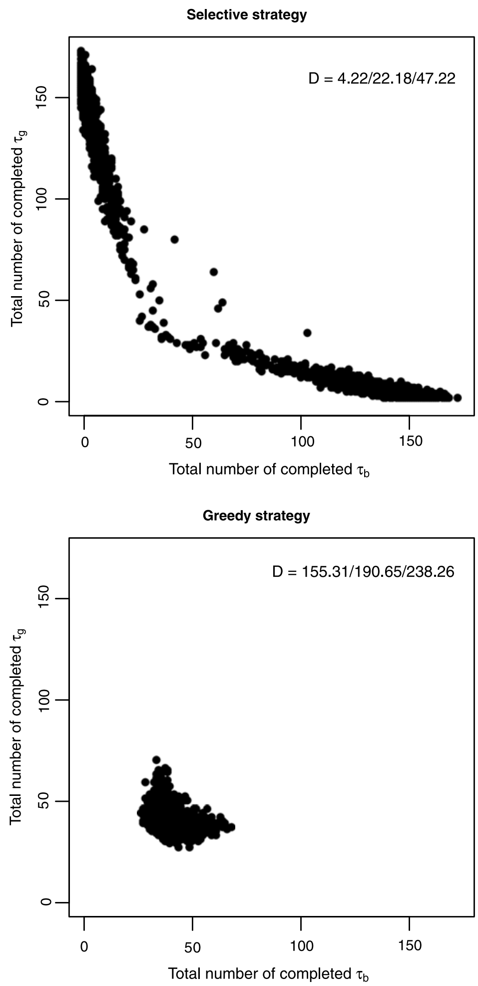


Fig. 8. Scatter plot of the total number of completed tasks per type for each robot, collected during the course of 20 experimental runs. Top: a swarm using the selective strategy effectively separates into two behaviorally distinct groups: one working mostly on τ_b, and another working mostly on τ_g. Bottom: The behavior of the greedy strategy is more homogeneous: the variability of the number of completed tasks of both types is much smaller than for the selective strategy. D indicates the value of the hierarchic social entropy; higher values of D indicate higher differentiation among the learning state of the robots of the swarm.

social entropy D as defined in Section 5.4. Fig. 8 (top) shows that when using the selective strategy, the swarm effectively separates into two behaviorally distinct groups: one working mostly on τ_b, and another working mostly on τ_g. Fig. 8 (bottom), on the other hand, shows that the behavior of the swarm using the greedy strategy is more uniform: the variability of the number of completed tasks of both types is much smaller than for the selective strategy. Comparing the hierarchic social entropy D, the robots using the selective strategy have a low hierarchic social entropy than the robots using the greedy strategy, which have a much higher hierarchic social entropy ($D = 4.22/22.18/47.22$ and $D = 155.30/190.65/238.25$, respectively). This reflects the degree of learning in the swarm as shown in Fig. 7: In case of the selective strategy, the robots fall into two distinct groups; differently, in case of the greedy strategy, random task allocation results in diverse learning states in the swarm.

Figs. 7 and 8 highlight the diversity and specialization observed in the swarm. The results explain the difference in performance observed in Fig. 6: even though all robots equally benefit from the advantages of learning, only the strategy that behaviorally specializes can successfully exploit these advantages.

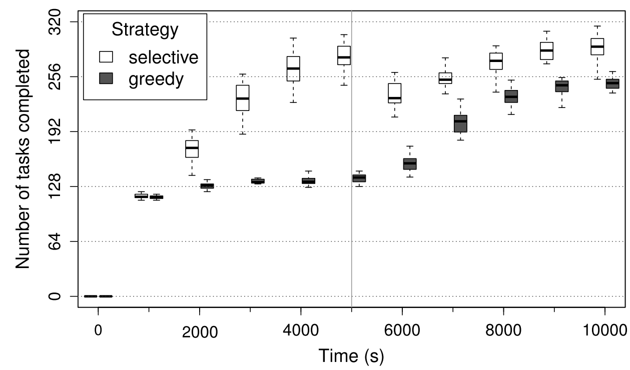


Fig. 9. Performance, defined as the total number of tasks completed in the preceding 1000 s, shown for each strategy over time. Performance is measured every 1000 s. Data collected over 20 experimental runs. The initial task ratio is $r = 0.5$, and is changed to 0.2 at $t = 5000$ s (gray vertical line).

6.2. Time-variant environment

In the second set of experiments we assess whether the benefits of behavioral specialization are affected by changes in the task ratio. To this end, we use an environment in which the task ratio changes over time. The initial task ratio is $r = 0.5$; when the experiment reaches half of its total duration, the task ratio is changed to $r = 0.2$, that is, tasks of type τ_g are predominant.

Again, we compare the performance of the selective strategy to the performance of the greedy strategy. Performance is defined as the amount of tasks completed in the preceding 1000 s; and we measure it every 1000 s. Fig. 9 shows the evolution of the performance of each strategy over time. In the plot, we report the median of the observation and the interquartile range (IQR). The observation of the performance for a given value of t is again distributed normally (see supplementary on-line material [30]); in the following we therefore report the mean(std). Comparing the performance of the two strategies, we can make the following observations. At $t = 5000$ s, the performance is 279.0(14.8) and 137.4(5.1) for the selective and the greedy strategy, respectively. This clearly replicates the results of the first experimental set. At the end of the experiment, on the other hand, the performance of the two strategies is 290.6(14.4) and 248.5(7.2), again for the selective and the greedy strategy, respectively. The results show that, at the end of the experiment, the selective strategy reaches a performance comparable to its performance before the change of the ratio, while the greedy strategy almost doubles its performance. In all cases with $t \geq 2000$, the performance of the two strategies is significantly different in favor of the selective strategy (Welch's t-test with $\alpha = 0.05$). We speculate that the increase in performance of the greedy strategy is due to the fact that one type of task is predominant in the environment. If a single task type is predominant in the environment, it is more likely that a robot working on random tasks works more often on the same type of task, thereby benefiting from the effect of learning.

In order to study the cause of the observed changes in performance, we evaluate the results by using the metrics F and P. Both measures are computed every 1000 s for the preceding 1000 s. The measures are not normally distributed (see on-line supplementary material [30]); we therefore report the interquartile range (IQR).

Fig. 10 (top) reports the specialization measure F for both strategies. The F measure confirms that robots using the selective strategy specialize well, as the value of F is close to 1 before the change of the task ratio. Note that F never reaches 1, which would be the case if there were no transitions at all. This is due to the fact that the selective strategy is based on a stochastic rule. Therefore, a robot may perform a task of one type even if it is fully

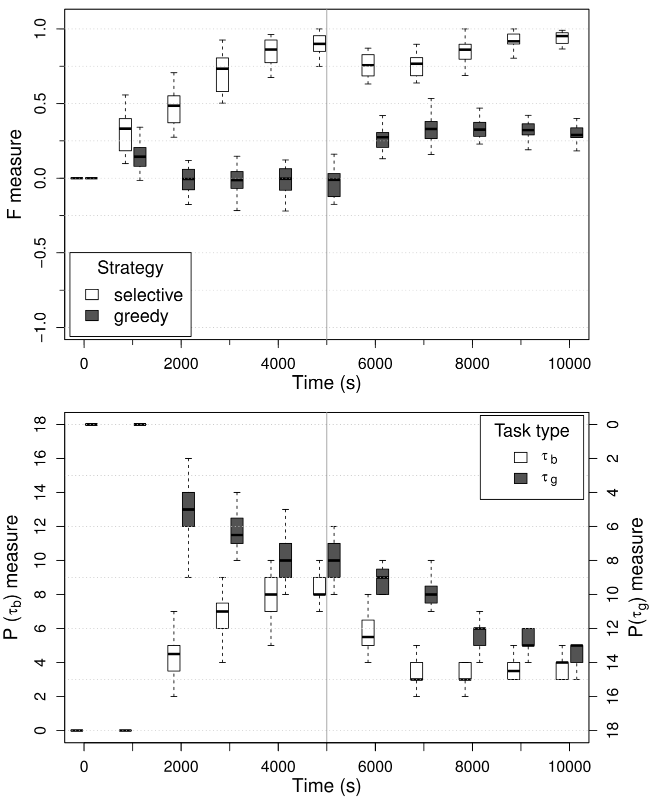


Fig. 10. Specialization in the swarm. Top: F measure for both strategies. Bottom: P measure for the selective strategy only. Both measures are computed every 1000 s for the preceding 1000 s. Data collected over 20 experimental runs. The initial task ratio is $r = 0.5$, and is changed to 0.2 at $t = 5000$ s (gray vertical line).

specialized in the other. After $t = 5000$ s the value of F changes: it decreases as the number of transitions per robot increases. This indicates that some of the robots using the selective strategy de-specialize from their current task type and specialize in the other type. At the end of the experiment, the swarm reaches again a high degree of specialization. Differently, the greedy strategy does not lead to specialization of the robots; the value of F is $\sim$0 for the first part of the experiment, that is, task allocation is random. The higher value of F for the greedy strategy in the second part of the experiment reflects the higher availability of tasks of type τ_g in the environment. As robots are more likely to encounter tasks of type τ_g, the number of transitions in the task sequence of a robot decreases, and the value of F increases in turn. This confirms our speculation above: the performance of the greedy strategy observed in Fig. 9 is a result of the reduced number of transitions, which lets the robots benefit more frequently from the effect of learning.

Fig. 10 (bottom) reports, for the selective strategy only, the number of robots specialized in the two types of tasks, using the measure P. The plot shows that before the change of the task ratio, approximately half of the swarm is specialized in one of the two task types and the other half on the other task type, which matches the task ratio. At time $t = 5000$ s the task ratio is changed to $r = 0.2$ in favor of τ_g. As it can be seen in the plot, some of the robots de-specialize from τ_b and subsequently specialize in τ_g. At the end of the experiment, the number of robots specialized in τ_b and τ_g is 4 and 14 respectively, again matching the task ratio.

In summary, we can say that the benefits of behavioral specialization depend on the distribution of the tasks in the environment.

6.3. *Periodically changing environments*

The results presented above indicate that changes in task ratio have a strong effect on the benefits of behavioral specialization. In the third experimental set, we therefore aim at studying this effect more closely by periodically changing the ratio of tasks as follows. Every Δt seconds, we alternate the task ratio between two values, r_1 and r_2. Δt is taken from the set $\{100, 1200, 5000\}$ s, and the two ratios r_1, r_2 are taken from the interval $[0.1, 0.9]$ in steps of 0.1. We only evaluate the cases in which $r_1 > r_2$; as we alternate between the two ratios, cases in which $r_1 < r_2$ would give analogous results. We conduct 20 experiments for each of the possible combinations of the parameters Δt, r_1, and r_2. Fig. 11 shows the result for environments that change frequently (every $\Delta t = 100$ s, left), moderately often (every $\Delta t = 1200$ s, middle) and rarely (every $\Delta t = 5000$ s, right).

Fig. 11 (left) shows that in case of frequent changes in task ratio, the selective strategy performs significantly better than the greedy strategy in all tested cases (Welch's t-test with $\alpha = 0.05$). This is due to the fact that the period between changes of the task ratio is shorter than the standard task completion time ($\Delta t < \overline{w}_{std}$). This results in an environment that remains effectively well-mixed: as changes in task ratio only have an effect after a task has been completed, all types of tasks are available in the environment at any given time. Thus, the robots using the selective strategy can fully exploit learning.

Fig. 11 (middle) shows the case in which the ratio changes every $\Delta t = 1200$ s $= 10\,\overline{w}_{std}$. This ensures that the ratio changes after most robots using the selective strategy have behaviorally specialized in a given task type. Therefore, among the three considered cases, this is the most difficult for the selective strategy. The plot shows that differences in performance between the selective strategy and the greedy strategy are generally smaller than in the case of $\Delta t = 100$ s. In case one of the two ratios is close to the extremes of task distribution (i.e., $r = 0$ or 1), the greedy strategy performs significantly better than the selective strategy. This suggests that in environments in which the task ratio can change abruptly from one extreme to the other, strategies using specialization are not advantageous.

Fig. 11 (right) shows the case in which the ratio changes every $\Delta t = 5000$ s, which is the setting we adopted in the set of experiments presented in Section 6.1 (for $r_1 = 0.5$ and $r_2 = 0.8$). The plot shows that the selective strategy is significantly better than the greedy strategy across the whole range of the ratio, excluding the extreme cases. This suggests that strategies using specialization are advantageous in environments that rarely change. This is consistent with the results presented in Section 6.1.

Upon analyzing all three plots in Fig. 11, we notice that they are approximately symmetric across the diagonal. This indicates that the impact of the ratio change on the performance of the selective strategy does not depend on the absolute value of the two ratios, but rather on the difference between the two ratios $|r_1 - r_2|$. This difference defines how many specialists need to re-specialize after a change so that the distribution of specialists matches again the distribution of tasks. The second factor that influences the difference in performance is the absolute distance of one of the ratios from the equal task distribution ($|r - 0.5|$). Differently from what we might expect, this is not due to an environment disadvantageous for the selective strategy. Quite on the contrary, the difference in performance is smaller because the greedy strategy benefits from the effect of learning in case one of the task types is more common than the other. As mentioned above, this effect can also be observed in Fig. 9.

In summary, we can conclude that behavioral specialization is advantageous in most cases of time-variant task distributions. Nevertheless, in the worst case, strategies employing behavioral

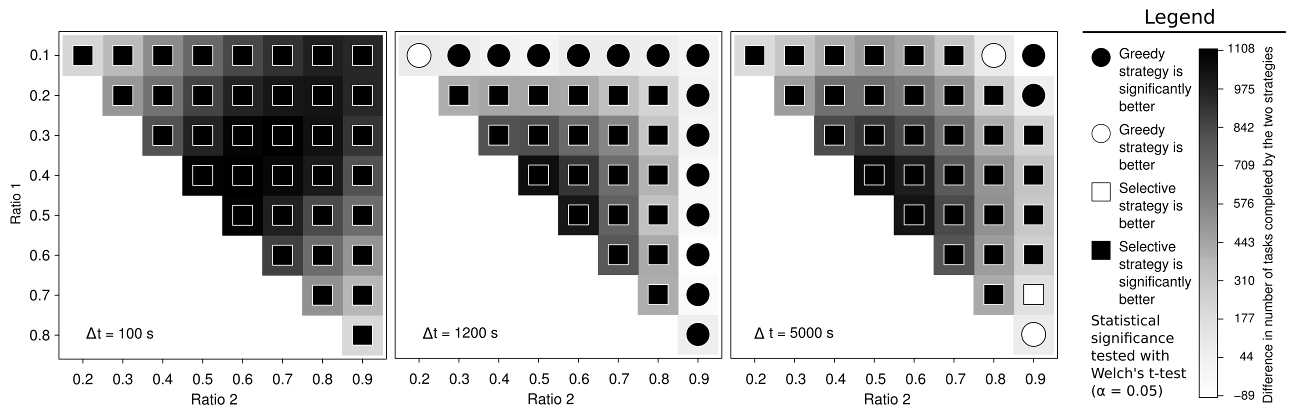


Fig. 11. Performance difference between the selective and the greedy strategy for environments in which the task ratio alternates between two values, r_1 and r_2. Task ratio changes every $\Delta t = 100$ s (left), 1200 s (middle), and 5000 s (right). The difference in performance is represented by shades of gray, with indication of which strategy is better and whether the difference is statistically significant or not (see symbols in the legend). Data collected over 20 experimental runs.

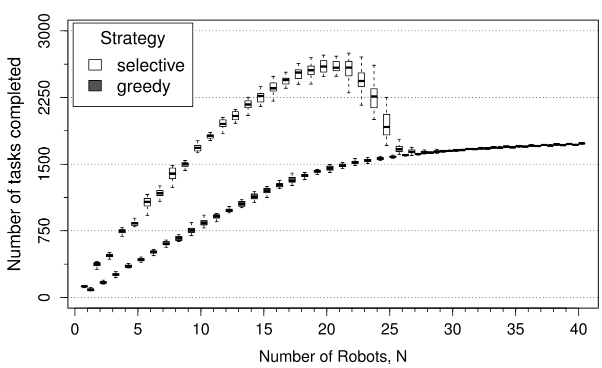


Fig. 12. Performance measured as the total number of tasks completed by the end of the experiment, shown for each strategy for different swarm sizes $N \in [1, 40]$. Data collected over 20 experimental runs.

specialization can exhibit drastic changes in performance. This confirms the speculation from our previous work [5] that behavioral specialization might be sensitive to changes in task distribution, and is prone to failure in the most extreme cases.

6.4. Scalability

In the next set of experiments we study whether a strategy exploiting behavioral specialization scales well. To this end, we run 20 experiments for each swarm size $N \in [1, 40]$ (in steps of 1) without changing the size of the arena, again for a duration of $t_{max} = 10{,}000$ simulated seconds each. Fig. 12 shows the total number of tasks completed by the two strategies. The swarm size has a strong impact on the performance of both strategies due to an effect commonly known as interference [31]: with increasing robot density, the individual performance decreases as robots increasingly interfere with each other. As a result, the performance of both strategies in Fig. 12 plateaus for a large number of individuals.

Additionally, the increasing number of robots of the swarm has a strong effect on the performance of the selective strategy even for smaller numbers of individuals. Fig. 12 shows that the performance peaks around $N = 20$ and reduces to the performance of the greedy strategy for swarm sizes $N > 24$. Considering that the total number of tasks concurrently available in the environment is $T = 24$, we speculate that the advantage of specialization depends on the relation between the swarm size and the number of concurrent tasks. As we did not test for different values of T, the evidence for such a dependency is non-conclusive and warrants for further examination. A possible explanation for such a dependency could be competition among robots: if competition for tasks is high, which is the case when there are more robots than tasks available in the environment, robots might go without working on a task for a long time, forgetting their behavioral specialization in the process.

The selective strategy fails gracefully because its worst performance is comparable with the performance of the greedy strategy. This result is in accordance with the finding presented by Li et al. [13].

In order to study the range of swarm sizes around the peak in performance more closely, Fig. 13 reports the F measure and the hierarchic social entropy D for $N \in [20, 30]$. As we can see in Fig. 13 (left), the value of F for robots using the selective strategy decreases for increasing N until it reaches the level of the greedy strategy. This indicates that the number of transitions in the task sequence increases for robots of larger swarms. This is most likely due to the fact that the robots forget experiences previously learned while continuously searching for tasks due to over-competition. This is confirmed by Fig. 13 (right): The diversity of the robots using the selective strategy increases around $N = 25$, which indicates that some of the robots can maintain specialization, while others cannot. For larger swarm sizes, the diversity of the robots using the selective strategy decreases again, as competition becomes so high that none of the robots of the swarm can specialize in a specific type of task.

6.5. Costs and benefits of specialization

In the last set of experiments we study whether specialization, which clearly has benefits in terms of task performance, entails costs that hinder the performance of the swarm. The assumption is that a robot specialized for a certain task spends more time searching for it. Thus, specialists might be less efficient than generalists, which spend less time searching for a suitable task to perform. In order to study the trade-off between costs and benefits, we vary the costs (search time of the robots) and benefits (the minimal task completion time in the maximal learned state). We vary the search time of the robots by changing the wheel speed s used while searching from 10% to 100% of the maximum speed s_{max}, in steps of 10%. This corresponds to changing the size of the environment and therefore the distance between tasks without affecting robot density, which would entail changes in performance due to interference [31]. We vary the minimal task completion time $\overline{w}_{min}$ from a minimum of 10% to a maximum of 100% of the standard task completion time $\overline{w}_{std}$, in steps of 10% ($k = \{10, 5, 3.34, 2.5, 2.0, 1.67, 1.43, 1.25, 1.11\}$). The observation of the total number of completed tasks at the end of the experiment

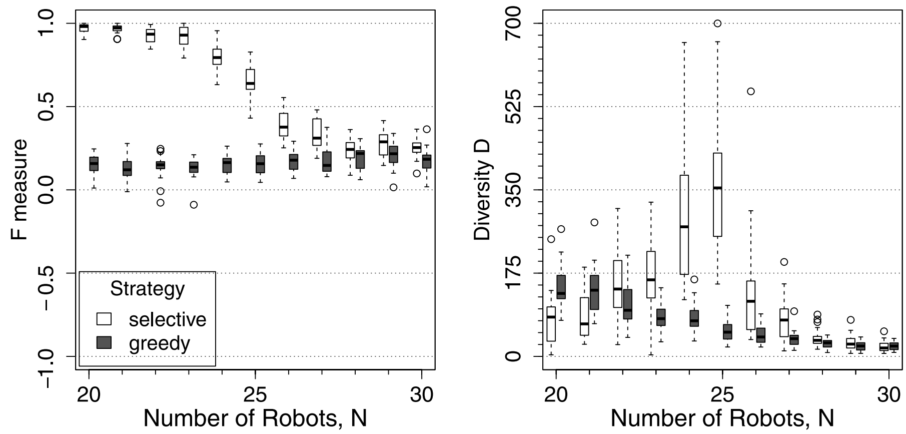


Fig. 13. Specialization and diversity shown for each strategy for different swarm sizes $N \in [20, 30]$, collected over 20 experimental runs. Left: F measure, indicating the number of transitions in the task sequence of a single robots. Right: hierarchic social entropy D, with higher values indicating higher diversity among the robots of the swarm.

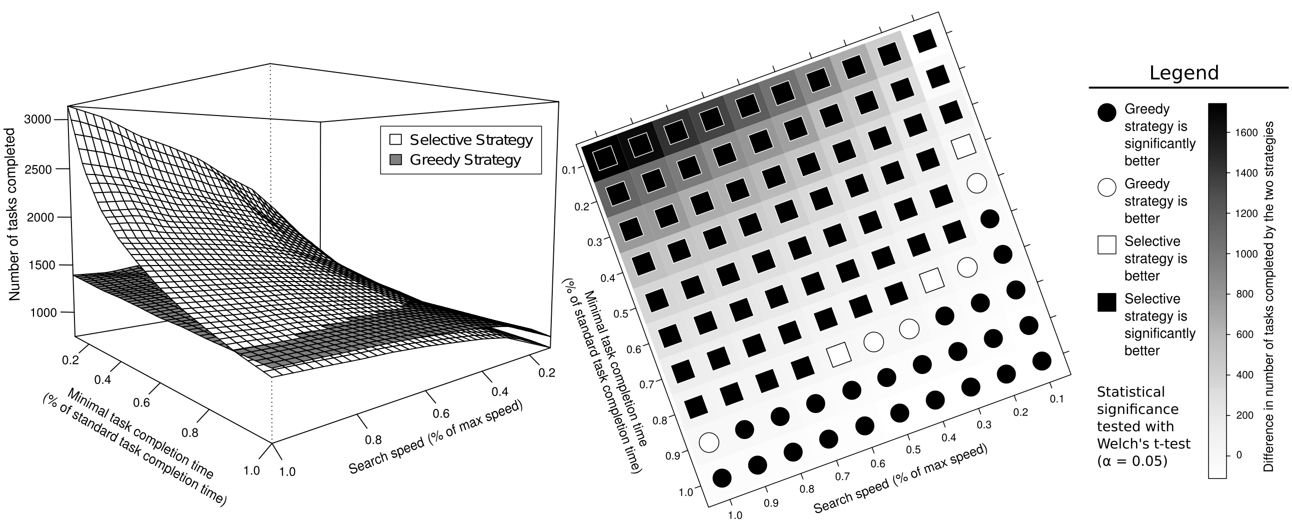


Fig. 14. Performance for different search speeds and task completion times at the maximal learning state, collected over 20 experimental runs. Left: observed mean of the number of completed tasks for the selective and greedy strategy (white and dark surface, respectively); Standard deviation $< 5\%$ for all tested cases (not shown). Right: difference in number of tasks completed by the two strategies (shades of gray), with indication of which strategy is better and whether the difference is statistically significant or not (see symbols in the legend).

is distributed normally (see supplementary on-line material [30]). The standard deviation is $< 5\%$ for all tested cases, therefore in the following we report only the mean. Fig. 14 (left) reports the mean of the total number of completed tasks, using the selective strategy (white surface) and the greedy strategy (dark surface), for different values of the search speed s and of the minimal task completion time $\overline{w}_{min}$. Fig. 14 (right) shows, for all combinations, the difference between the mean number of tasks completed by the two strategies by the end of the experiment (shades of gray), and if this difference is statistically significant or not (see symbols in the legend).

The plot on the left of Fig. 14 shows that the greedy strategy is less affected by changes of the two parameters. The change of the minimal task completion time has almost no effect on the performance of the greedy strategy as it does not behaviorally specialize and thus does not benefit from the effects of learning in a consistent manner. Moreover, the performance of the greedy strategy is only slightly affected by the search speed as the number of tasks concurrently available in the environment remains constant and robots using this strategy accept every task they encounter. The performance of the selective strategy, on the other hand, varies considerably in relation to the value of the two parameters, highlighting costs and benefits of specialization. The plot on the right shows that when the minimal task completion time $\overline{w}_{min}$ is greater than 80% of the $\overline{w}_{std}$ or the wheel speed s is 10% of the maximum speed s_{max}, the greedy strategy performs better than the selective strategy.

This confirms our assumption that robots specializing in a certain task are prone to losing efficiency due to high costs of behavioral specialization, for example, longer search times. Behavioral specialization is therefore not to be considered in terms of benefits only, as it is affected by external factors such as task availability and the spatial distribution of the tasks, which might lower its benefits considerably.

7. Conclusions

Behavioral specialization is common in the organization of large groups of individuals such as humans or social insects, as it has many advantages, an important one being that it allows individuals to exploit improvements in task performance due to learning. However, specialization can also entail costs: specialists may need to spend more time searching for their tasks than generalists.

To study behavioral specialization, we considered a system in which robots can perform two types of tasks, available in the environment with spatial and temporal distributions that are unknown to the robots. Robots improve their task performance upon repetition: this simplified learning model allows us to draw general conclusions on behavioral specialization without the need of implementing an actual learning technique.

We employed two simple task allocation strategies to study the system: a strategy in which robots select among the available tasks in order to exploit learning by behaviorally specializing in a certain type of task, and another strategy in which task allocation is random. We studied the system in simulation-based experiments, focusing on its response to changes in the distribution of task types. Additionally, we studied its behavior under various conditions that affect the costs and benefits of specialization. Results indicate that spatial effects, such as interference among robots, have a major influence on the costs and benefits of specialization. We identified cases in which the costs of specialization overcome its benefits. A task allocation strategy that does not use specialization is preferable in these cases. The results also suggest that behavioral specialization is not advantageous in environments that are highly time-variant, as specialists may not be fast enough to adapt to changes in the distribution of tasks.

There are several possible directions for future research. One is the study of specialization in swarms of heterogeneous robots, where benefits and costs of specialization are linked to morphological differences between robots. These differences can be explicit (e.g., different capabilities or equipment) or implicit (e.g., heterogeneity due to production tolerances of the hardware [17]). Another direction is the implementation of learning as an actual improvement of the task-related performance of the robots as opposed to the modeling of the improvement in an abstract way as presented in this paper. A third possibility for future research is to study the influence of different characteristics of the task types on the benefits of specialization, such as tasks that exhibit dependency among them (e.g., task types that require a certain order of execution or task types that require concurrent cooperation of multiple robots, similar to the study presented by Li et al. [13]).

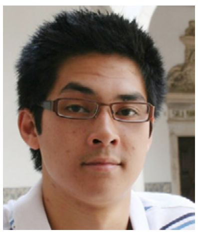
Acknowledgments

The research leading to the results presented in this paper has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement no. 246939. Marco Dorigo, Mauro Birattari, and Arne Brutschy acknowledge the support from the Belgian F.R.S.–FNRS. Giovanni Pini acknowledges the support from Université Libre de Bruxelles through the "Fonds David & Alice Van Buuren". Marco Frison acknowledges the support from "Seconda Facoltà di Ingegneria", Alma Mater Studiorum, Università di Bologna. Andrea Roli acknowledges the support from the "Brains (Back) to Brussels 2009" program funded by IRSIB-Institut d'Encouragement de la Recherche Scientifique et de l'Innovation de Bruxelles.

References

[1] S.N. Beshers, J.H. Fewell, Models of division of labor in social insects, Annual Review of Entomology 46 (2001) 413–440.
[2] S. Garnier, J. Gautrais, G. Theraulaz, The biological principles of swarm intelligence, Swarm Intelligence 1 (2007) 3–31.
[3] G. Nitschke, M. Schut, A. Eiben, Emergent specialization in biologically inspired collective behavior systems, in: Intelligent Complex Adaptive Systems, IGI Publishing, New York, 2007, pp. 215–253. (Chapter 8).
[4] F.L.W. Ratnieks, C. Anderson, Task partitioning in insect societies, Insectes Sociaux 46 (2) (1999) 95–108.
[5] A. Brutschy, N.-L. Tran, N. Baiboun, M. Frison, G. Pini, A. Roli, M. Dorigo, M. Birattari, Costs and benefits of behavioral specialization, in: Towards Autonomous Robotic Systems—12th Annual Conference, TAROS 2011, in: LNCS, vol. 6856, Springer, Berlin/Heidelberg, Germany, 2011, pp. 90–101.
[6] E. Bonabeau, M. Dorigo, G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, New York, 1999.
[7] J. Gautrais, G. Theraulaz, J.-L. Deneubourg, C. Anderson, Emergent polyethism as a consequence of increased colony size in insect societies, Journal of Theoretical Biology 215 (3) (2002) 363–373.
[8] K. Diwold, A. Scheidler, M. Middendorf, The effect of spatial organisation in response threshold models for social insects, in: European Conference on Complex Systems, ECCS'09, University of Warwick, Warwick, UK, 2009, pp. 21–25.
[9] T.O. Richardson, K. Christensen, N.R. Franks, H.J. Jensen, A.B. Sendova-Franks, Ants in a labyrinth: a statistical mechanics approach to the division of labour, PLoS ONE 6 (4) (2011) e18416.
[10] M. Brooks, Analogies from biology: distributed sensing and learning, in: J.T. Tou, J.G. Balchen (Eds.), NATO Advanced Research Workshop on Highly Redundant Sensing in Robotic Systems, Springer, Berlin/Heidelberg, Germany, 1990, pp. 92–102.
[11] A. Dornhaus, Specialization does not predict individual efficiency in an ant, PLoS Biology 6 (11) (2008) e285.
[12] L. Panait, S. Luke, Cooperative multi-agent learning: the state of the art, Autonomous Agents and Multi-Agent Systems 11 (3) (2005) 387–434.
[13] L. Li, A. Martinoli, Y.S. Abu-Mostafa, Learning and measuring specialization in collaborative swarm systems, Adaptive Behavior 12 (3–4) (2004) 199–212.
[14] M. Hsieh, A. Halasz, E. Cubuk, S. Schoenholz, A. Martinoli, Specialization as an optimal strategy under varying external conditions, in: IEEE International Conference on Robotics and Automation 2009, ICRA'09, IEEE Press, Piscataway, NJ, 2009, pp. 1941–1946.
[15] C. Jones, M.J. Matarić, Adaptive division of labor in large-scale minimalist multi-robot systems, in: 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2003, IEEE Press, Piscataway, NJ, 2003, pp. 1969–1974.
[16] A. Murciano, J.R. Millán, J. Zamora, Specialization in multi-agent systems through learning, Biological Cybernetics 76 (5) (1997) 375–382.
[17] T.H. Labella, M. Dorigo, J.L. Deneubourg, Division of labour in a group of robots inspired by ants' foraging behaviour, ACM Transactions on Autonomous and Adaptive Systems 1 (1) (2006) 4–25.
[18] W. Liu, A.F.T. Winfield, J. Sa, J. Chen, L. Dou, Towards energy optimization: emergent task allocation in a swarm of foraging robots, Adaptive Behavior 15 (3) (2007) 289–305.
[19] Y. Ikemoto, T. Miura, H. Asama, Adaptive division-of-labor control algorithm for multi-robot systems, Journal of Robotics and Mechatronics 22 (4) (2010) 514–525.
[20] B.P. Gerkey, M.J. Matarić, A formal framework for the study of task allocation in multi-robot systems, International Journal of Robotics Research 23 (9) (2004) 939–954.
[21] Z. Kira, R.C. Arkin, Forgetting bad behavior: memory management for case-based navigation, in: Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004, IEEE Press, Piscataway, NJ, 2004, pp. 3145–3152.
[22] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.C. Zufferey, D. Floreano, A. Martinoli, The *e*-puck, a robot designed for education in engineering, in: P.J.S. Gonçalves, P.J.D. Torres, C.M.O. Alves (Eds.), Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, IPCB: Instituto Politècnico de Castelo Branco, Castelo Branco, Portugal, 2009, pp. 59–65.
[23] A. Brutschy, G. Pini, N. Baiboun, A. Decugnière, M. Birattari, The TAM: A device for task abstraction for the *e*-puck robot, Tech. Rep. TR/IRIDIA/2010-015, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, 2010.
[24] G. Pini, A. Brutschy, M. Frison, A. Roli, M. Birattari, M. Dorigo, Task partitioning in swarms of robots: an adaptive method for strategy selection, Swarm Intelligence 5 (3–4) (2011) 283–304.
[25] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, T. Stirling, A. Gutiérrez, L.M. Gambardella, M. Dorigo, ARGoS: a modular, multi-engine simulator for heterogeneous swarm robotics, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2011, IEEE Computer Society Press, Los Alamitos, CA, 2011, pp. 5027–5034.
[26] M. Dorigo, D. Floreano, L.M. Gambardella, F. Mondada, S. Nolfi, T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, D. Burnier, A. Campo, A.L. Christensen, A. Decugnière, G.D. Caro, F. Ducatelle, E. Ferrante, A. Förster, J.M. Gonzales, J. Guzzi, V. Longchamp, S. Magnenat, N. Mathews, M.M. de Oca, R. O'Grady, C. Pinciroli, G. Pini, P. Rétornaz, J. Roberts, V. Sperati, T. Stirling, A. Stranieri, T. Stützle, V. Trianni, E. Tuci, A.E. Turgut, F. Vaussard, Swarmanoid: a novel concept for the study of heterogeneous robotic swarms, IEEE Robotics & Automation Magazine, 2012 (in press).
[27] T. Balch, Hierarchic social entropy: an information theoretic measure of robot group diversity, Autonomous Robots 8 (2000) 209–237.
[28] C.E. Shannon, The Mathematical Theory of Communication, University of Illinois Press, Champaign, IL, 1949.
[29] S. O'Donnell, R.L. Jeanne, Forager specialization and the control of nest repair in polybia occidentalis olivier (hymenoptera: vespidae), Behavioral Ecology and Sociobiology 27 (5) (1990) 359–364.
[30] A. Brutschy, N.-L. Tran, N. Baiboun, M. Frison, G. Pini, A. Roli, M. Dorigo, M. Birattari, Costs and benefits of behavioral specialization—Online supplementary material, 2011, http://iridia.ulb.ac.be/supp/IridiaSupp2011-024/.
[31] G. Pini, A. Brutschy, M. Birattari, M. Dorigo, Task partitioning in swarms of robots: reducing performance losses due to interference at shared resources, in: J.A. Cetto, J. Filipe, J.-L. Ferrier (Eds.), Informatics in Control, Automation and Robotics, in: LNEE, vol. 85, Springer, Berlin/Heidelberg, Germany, 2011, pp. 217–228.

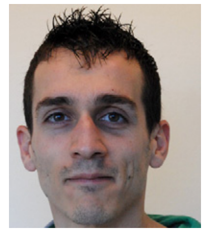
Arne Brutschy received a Diploma in Computer Science from the University of Leipzig, Germany. He is currently a Ph.D. candidate in Applied Sciences at IRIDIA, Université Libre de Bruxelles, Belgium. Arne Brutschy received a scholarship from the fund for scientific research F.R.S.-FNRS of Belgium's French Community for the duration of his Ph.D studies. His research interests are in artificial intelligence and swarm robotics, with a focus on self-organized task allocation and task partitioning.



Nam-Luc Tran has received a Master's degree in Biomedical Engineering from the Universit Libre de Bruxelles, Belgium. He did his Master's thesis in Robotics and Swarm Intelligence at IRIDIA, Université Libre de Bruxelles, Belgium. Nam-Luc Tran now works as a research and development engineer at Euranova, an IT expert consultancy company. His interests cover distributed systems, high availability and high scalability systems.

Nadir Baiboun received a Master's degree in automation engineering from ECAM, Institut Supérieur Industriel, Belgium. His Master's thesis was done on the topic of robotics and swarm intelligence at IRIDIA, Université Libre de Bruxelles, Belgium. Nadir Baiboun is currently working as a research engineer at CERDECAM, a research and development unit affiliated with ECAM. His research interests focus on simulation of batteries, as well as embedded systems.

Marco Frison received a Master's degree in computer engineering from the University of Bologna, Italy. He carried out his Master's thesis about adaptive task partitioning and swarm robotics at IRIDIA, Université Libre de Bruxelles, Belgium. Marco Frison currently works as a freelance security consultant for major Italian competitors in the banking, insurance, telecommunications, and energy sector. His interests spread from penetration testing to lock-picking.



Giovanni Pini is a Ph.D. candidate in applied sciences at IRIDIA, Université libre de Bruxelles, Belgium. He holds a Master's degree in computer science engineering, received from Politecnico di Milano, Italy, in 2007. His research work focuses on swarm intelligence and swarm robotics, and self-organized task-partitioning and task-allocation.

Andrea Roli received the Ph.D. degree in computer science and electronic engineering from Alma Mater Studiorum Université di Bologna, where he is an assistant professor. He teaches subjects in artificial intelligence, complex systems and computer science basics. His main current research interests include metaheuristics and complex systems, with applications to swarm intelligence, bioinformatics and genetic regulatory network models. Andrea Roli is a member of the steering committee of the Italian Association for Artificial Intelligence.

Marco Dorigo received his Ph.D. in electronic engineering in 1992 from Politecnico di Milano, Italy, and the title of Agrégé de l'Enseignement Supérieur, from ULB, in 1995. Since 1996, he has been a tenured Researcher of the fund for scientific research F.R.S.-FNRS of Belgium's French Community, and a Research Director of IRIDIA, ULB. He is the inventor of the ant colony optimization metaheuristic. His current research interests include swarm intelligence, swarm robotics, and metaheuristics for discrete optimization. He is the Editor-in-Chief of Swarm Intelligence. Dr. Dorigo is a Fellow of the IEEE and of ECCAI. He was awarded the Italian Prize for Artificial Intelligence in 1996, the Marie Curie Excellence Award in 2003, the Dr. A. De Leeuw–Damry–Bourlart award in applied sciences in 2005, the Cajastur International Prize for Soft Computing in 2007, and an ERC Advanced Grant in 2010.

Mauro Birattari received his Master's degree in electronic engineering from Politecnico di Milano, Italy, in 1997; and his Ph.D. degree in information technologies from Université Libre de Bruxelles, Belgium, in 2004. He is currently with IRIDIA, Université Libre de Bruxelles, as a research associate of the fund for scientific research F.R.S.-FNRS of Belgium's French Community. Dr. Birattari co-authored about 100 peer-reviewed scientific publications in the field of computational intelligence. Dr. Birattari is an associate editor for the journal Swarm Intelligence, and an area editor for the journal Computers & Industrial Engineering.

Please cite this article in press as: A. Brutschy, et al., Costs and benefits of behavioral specialization, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.05.018