Self-Organised Task Allocation in a Group of Robots

Thomas H. Labella, Marco Dorigo and
Jean-Louis Deneubourg

Technical Report No.
TR/IRIDIA/2004-6
November 30, 2004

Published in R. Alami, editor, *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS04)*,
Toulouse, France, June 23–25, 2004

Self-Organised Task Allocation in a Group of Robots

Thomas H. Labella, Marco Dorigo and Jean-Louis Deneubourg

No Institute Given

Summary. Robot foraging, a frequently used test application for collective robotics, consists in a group of robots retrieving a set of opportunely defined objects to a target location. A commonly observed experimental result is that the retrieving efficiency of the group of robots, measured for example as the number of units retrieved by a robot in a given time interval, tends to decrease with increasing group sizes. In this paper we describe a biology inspired method for tuning the number of foraging robots in order to improve the group efficiency. As a result of our experiments, in which robots use only locally available information and do not communicate with each other, we observe self-organised task allocation. This task allocation is effective in exploiting mechanical differences among the robots inducing specialisation in the robots activities.

1 Introduction

The interest for collective robotics of scientists from disciplines as different as biology and engineering has recently been increasing. For instance, biologists have started to use robots for testing their theories about social animals, while engineers see in collective robotics a means for finding solutions to problems that cannot be solved efficiently by a single robot.

In this paper we consider a typical problem in collective robotics: foraging. Robot foraging consists in the cooperative activity of a group of robots whose goal is to retrieve to a target location a set of opportunely defined objects. A well known problem in robot foraging is the reduction in the performance of the group of robots, measured for example as the number of units retrieved by a robot in a given time interval, when the group size becomes bigger because of interferences among robots (Goldberg and Matarić, 1997; Balch, 1999).

A possible solution to this decreasing efficiency problem is to adopt some task allocation[1] mechanism that allows to automatically determine the optimal size of a group of robots that cooperate in a foraging application.

---

[1] In the collective robotics literature, the term “task” is given two different meanings, depending on whether the common goal involves one or more tasks: in the first case, which is also ours, a task allocation method is meant to find the optimal num-

We propose a method inspired by biology to tune the number of foragers. This method, that exploits positive and negative feedbacks as typically done by self-organised systems (Camazine et al., 2001), does not use any form of direct or symbolic communication and does not require human intervention.

Our work is part of the SWARM-BOTS project,[2] whose aim is to develop a new robotic system, a *swarm-bot*, composed of several independent and small modules, called *s-bots.* Each module is autonomous and capable of connecting to other modules to self-assemble into a *swarm-bot.* The control program of each *s-bot* exploits techniques derived from swarm intelligence studies (Bonabeau et al., 1999) and collaboration among the *s-bots* is achieved by means of stigmergic communication (Grassé, 1959; Dorigo et al., 2000).

This paper is organised as follows. Section 2 introduces the problem of foraging, its issues, and illustrates the biological inspiration of our work. Section 3 describes the hardware and the software used in our experiments. Section 4 shows and analyses the results. Section 5 summarises related work, and finally, Section 6 draws some conclusions.

2 Problem Description and Issues

Foraging is considered a canonical test domain for collective robotics (Cao et al., 1997). The terminology we use in this paper is borrowed from biology: for instance we use the term “prey retrieval” as a synonymous for “retrieving an object”.

The environment in which a prey retrieval experiment is performed includes: a group of robots, also called a “colony”; objects spread in the environment, called “prey”, that may have different sizes or weights, may be fixed or moving, and may appear and disappear randomly; and a special area called “nest”. The robots goal is to collect the prey and bring them to the nest.

A colony of robots can solve the problem in shorter time than a single robot, but the efficiency degrades when there are too many robots because of negative interferences. One way to avoid this is to choose how many robots should be engaged in prey retrieval in such a way that the efficiency of the group is maximised. In other words, to use task allocation.

Biologists have developed numerous models to explain how task allocation can be achieved without using direct communication. In this paper, we refer to Deneubourg et al.’s (1987) model, in which the individuals adapt and learn during their life-time. In Deneubourg et al.’s model, each ant is an agent that leaves the nest with probability P_l. If its foraging trip is successful, that is, the ant retrieves a prey, it increases its P_l by a constant Δ. If it is not successful, it decreases it by the same constant. Unfortunately, the authors performed tests only with numerical simulations.

---

ber of robots to perform the given task (as in Krieger and Billeter, 2000; Agassounon and Martinoli, 2002); in the second case, task allocation is in charge of assigning one robot to each task (as in Gerkey and Matarić, 2003).

[2] For more information on the project see `www.swarm-bots.org`

The aim of this paper is to assess the feasibility of using similar mechanism to control a group of robots and to test whether this form of adaptation, which was only shown to be theoretically correct, works in the real world. Therefore, we decided to use real robots and not simulation for the first experiments. Time and technical constraints obliged us also to use only one colony size, leaving as future work the study of further aspects, such as the dependencies on group size and on the prey density.

In a previous work (Labella et al., 2004), we showed that a variant of this model, described in Section 3, can improve the efficiency of the colony by reducing the number of robots looking for prey. Here, we show that the improvement is achieved by means of group-level task allocation which increases P_l in some robots in the colony and decreases it in the remaining ones (Section 4.1). Additionally, in Section 4.2 we show that our allocation mechanism tends to consistently select the same robots to be foragers, which means that the allocation mechanism exploits differences among the robots that make some of them more suited for prey retrieval. The differences we refer to are not intentionally implemented in the robots, but come from the fact that two artifacts can never be perfectly identical.[3] The mechanism that we propose is not based on direct communication among robots, is completely decentralised and does not require human intervention. It can therefore be considered as *self-organised*.

3 Hardware and Software

The *s-bots* were still in the prototyping phase at the time of the experiments. Therefore, we decided to run the experiments using robots built with Lego Mindstorms$^{\text{TM}}$. The resulting robot, that we name *MindS-bot*, is presented in Fig. 1(a). *MindS-bots* use tracks to move. They have two arms, placed symmetrically with respect to the centre of the robot, that they use to grip the prey. Two light sensors are on the top of the *MindS-bot*: one on the front, which is used to sense prey, and one on the back, which is used to search for and go to the nest. Two bumpers, which are also placed on the front and on the back side, are used to avoid obstacles.

Figure 1(b) depicts the control program of the *MindS-bots*. Different states deal with the different phases of prey retrieval, as follows:

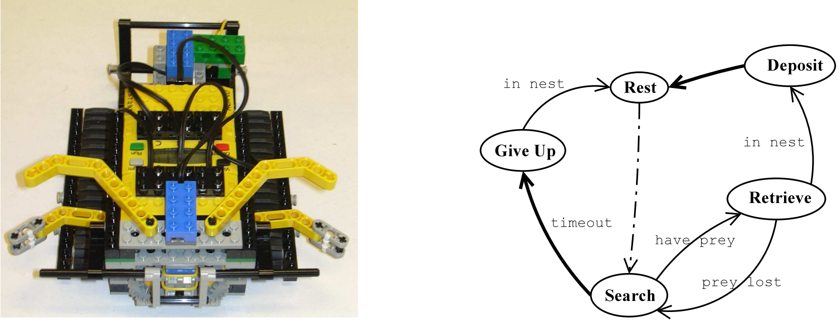
Search: the *MindS-bot* looks for a prey and avoids collisions with other *MindS-bots*. If a prey is found, the *MindS-bot* grasps it. If it has spent too much time searching for a prey without finding any, it gives up.

Retrieve: the *MindS-bot* looks for the nest and pulls the prey toward it.

Deposit: the *MindS-bot* leaves the prey in the nest and positions itself for the next foraging trip.

---

[3] For instance, we observed during the experiments that the motors of some robots were faster and that some robots could grasp stronger than others, although the motors were the same models and the robots were built in the same way.



(a) Front view of a *MindS-bot* (b) Schema of the control system

Fig. 1. Hardware and software of a *MindS-bot*. In (b), states represent different phases of the retrieval. The labels on each edge represent the conditions that let the transitions to other states occur. The bold edges show when the probability to leave the nest is updated. The edge between **Rest** and **Search** is dash-dotted to indicate that the transition occurs probabilistically with probability P_l.

Algorithm 1 Variable Delta Rule (VDR). P_l is the probability to leave the nest, *succ* and $fail$ are the number of consecutive successes and failures.

```
initialisation: $\mathrm{succ} \leftarrow 0; \mathrm{fail} \leftarrow 0; P_\mathrm{l} \leftarrow P_\mathrm{init}$
if success then
    $\mathrm{succ} \leftarrow \mathrm{succ} + 1; \mathrm{fail} \leftarrow 0$
    $P_\mathrm{l} \leftarrow min\{P_\mathrm{max}, P_\mathrm{l} + \mathrm{succ} \cdot \Delta\}$
end if
if failure then
    $\mathrm{succ} \leftarrow 0; \mathrm{fail} \leftarrow \mathrm{fail} + 1$
    $P_\mathrm{l} \leftarrow max\{P_\mathrm{min}, P_\mathrm{l} - \mathrm{fail} \cdot \Delta\}$
end if
```

Give Up: the *MindS-bot* looks for the nest and returns to it.
Rest: the *MindS-bot* rests in the nest before restarting the search.

Transitions between states occur on the basis of events that are either external (e.g., finding a prey or entering the nest) or internal to the robot (e.g., a timeout). The labels on the edges in the graph of Fig. 1(b) show the conditions that must be true for the transitions to occur.

The *MindS-bots* change from **Rest** to **Search** with probability P_l, whose value is updated during the transitions from **Search** to **Give Up** (henceforth called *failure*) and from **Deposit** to **Rest** (henceforth called *success*). The update is done as shown in Algorithm 1, named Variable Delta Rule (VDR). The algorithm increments or decrements P_l by a constant Δ multiplied by the number of consecutive successes or failures (not present in the model of Deneubourg et al.). It then bounds P_l in the range $[P_\mathrm{min}, P_\mathrm{max}]$.

4 Experiments and Results

We performed ten experiments using a circular arena (Figure 2) with a diameter of 2.40 m. Each experiment lasted 2400 s. A light bulb was placed over

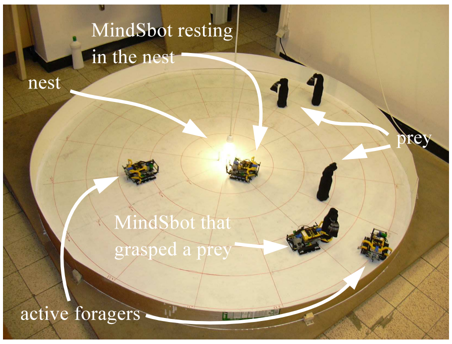


Fig. 2. Snapshot of an experiment. Four *MindS-bots* are looking for three prey. The nest is indicated by a light in the centre. One *MindS-bot* is resting in the nest, two are exploring the environment, and the fourth is retrieving a prey to the nest.

the centre of the nest area. Walls and floor were white painted to be more reflective, prey were black cylinders.

The timeout was set to 228 s.[4] P_{max} was set to 0.05, which corresponds to a mean idleness time in the nest of 20 s. P_{min} was 0.0015 (mean idleness time: 666.6 s) and $\varDelta$ was 0.005. At the beginning of each experiment P_l was set to 0.033 (mean idleness time: 30 s). These values were chosen on the basis of a trial-and-error methodology. Prey appeared randomly in the arena with probability 0.006 per second. Their position was selected randomly to be between 0.5 m and 1.1 m from the centre of the arena.[5] The colony size was of four *MindS-bots*, selected out of a pool of six, and some of them were substituted after each experiment.

The next section analyses the task allocation that occurs in the colony, while Sec. 4.2 shows how task allocation takes into account mechanical differences among *MindS-bots*.

4.1 Task allocation

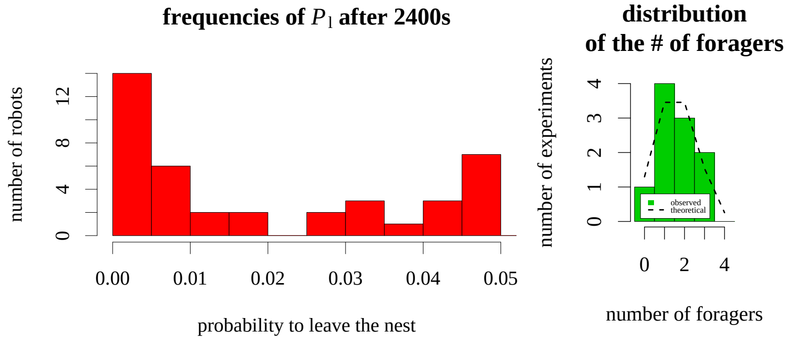
At any given instant t after the beginning of the experiment, the value of P_l in a *MindS-bot* is a random variable. Whether task allocation occurs or not can be observed in the distribution of P_l: if task allocation occurs, then at the end of the experiments some of the *MindS-bots* will have high P_l while the others will have a low P_l, and the distribution of P_l will present two peaks; otherwise it will have only one peak.

We recorded the value of P_l for each *MindS-bot* during the experiments and estimated the distribution. Figure 3(a) shows the result after 2400 s and its two-peak shape confirms that task allocation has occurred. We classify

---

[4] This value is the estimate of the median time needed by one *MindS-bot* to find one prey when it is alone in the arena.

[5] A computer placed next to the arena was used to warn the experimenter, by means of a random number generator, when and where a new prey should appear.



(a) (b)

Fig. 3. (a) Frequency of P_l observed 2400 s after the beginning of experiments. The two peaks demonstrate the occurrence of task allocation. We classify the *MindS-bots* in two groups using 0.025 as a threshold: 40% of the observations are above it. (b) Distribution of the number of foragers ($P_l > 0.025$) observed in each experiment compared with the theoretical binomial distribution with $p = 0.4$.

therefore the *MindS-bots* in two groups: those with P_l higher than 0.025 are called *foragers*, while those with P_l lower than 0.025 are called *loafers*.

One might object that the peak on the right of the distribution in Figure 3(a) could be the result of a few experiments in which all the *MindS-bots* happen to be foragers. To see that this is not the case, it is enough to look at the number of foragers which were present in each experiment, and how this number is distributed. From the data in Figure 3(a), we know that 40% of the population are foragers. Therefore, we expect that the number of foragers in each experiment follows a binomial distribution with $p = 0.4$. Figure 3(b) shows that the profiles of the theoretical and the observed distributions are very similar and suggests that further experiments will confirm the matching.

It is interesting to note that both positive and negative feedbacks are present in the colony. Positive feedback is given by the fact that the higher the P_l of a *MindS-bot*, the shorter the time the *MindS-bot* remains in the nest and, therefore, the shorter the time until it finds a new prey and increases its P_l again. Negative feedback is given by competition among *MindS-bot*: every prey taken by one *MindS-bot* decreases the probability that the others can successfully retrieve. These two forms of feedback are likely to contribute to the occurrence of task allocation.

The evolution of the distribution of P_l over time (Figure 4) shows that the group of foragers starts forming later than the group of loafers (the former at 1500 s, the latter at 1000 s). At the beginning, some *MindS-bots* become loafers because they are not successful, while the others alternate successes with failures. The fewer *MindS-bots* are foraging, the fewer competitors are present and the higher is the probability that the foraging *MindS-bots* will remain foragers.

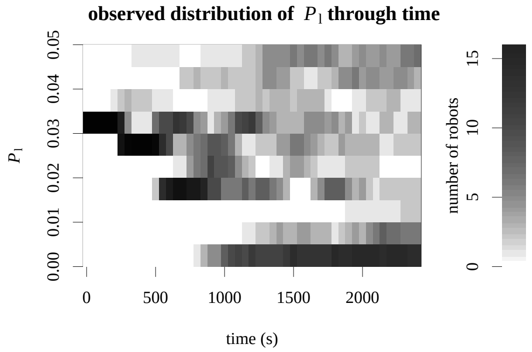


Fig. 4. Dynamics of the observed frequency of P_l. The darkness of a cell in position (t, p) is proportional to the number of *MindS-bots* with $p = P_l$ after t seconds from the beginning of the experiment. The relationship is given by the bar on the right. At $t = 0$ all the *MindS-bots* have $P_l = 0.033$ (see the black stripe on the left). After 1000 s the number of *MindS-bots* with low P_l (the loafers) drastically increases (see the dark stripe on the bottom). Similarly, after 1500 s, the number of robots with high P_l (the foragers) increases, although slowly and reaching a lower value than the loafers (top-right part of the plot).

4.2 Exploitation of mechanical differences

Given the stochastic nature of the experiments, we can model the fact that a given *MindS-bot* i is a forager at the end of an experiment as a random event. As in our experiments we use groups of 4 robots selected out of a pool of $N = 6$ robots, the probability of this random event may depend on the specific group G_k, $k \in \{1, \ldots, \binom{6}{4}\}$, to which i belongs in a given experiment: we denote this probability by $P_f(i|k)$.

There are two possibilities, depending on whether the following condition is true or not:

$$\exists\, i, k, j : \quad P_f(i|k) \neq P_f(i|j), \quad k \neq j \ . \tag{1}$$

If (1) is true, then the allocation mechanism exploits mechanical differences, which is what we want to prove. On the contrary, if (1) is false, then $P_f(i|k) = P_f(i)$ (that is, the probability of i being a forager is not a function of the group G_k to which it belongs) and we have that either the following condition is true:

$$\exists\, i, j : \quad i \neq j, P_f(i) \neq P_f(j) \ , \tag{2a}$$

in which case, once again, the allocation mechanism exploits mechanical differences, or the following equation is true:

$$P_f(i) = P_f(j) \quad \forall\, i, j \in \{1, \ldots, N\} \ , \tag{2b}$$

in which case the allocation mechanism does not exploit mechanical differences (note that (2a) and (2b) are mutually exclusive). If we assume that (1) is false, we can show that also (2b) is false considering the data in Table 1, which

Table 1. For each robot, identified by an unique name, the total number of experiments in which it was used and the number of times it was a forager are reported. Data refers to ten experiment, four *MindS-bots* per experiment.

ID	Tot. Exp.	$\#_{foragers}$
*MindS-bot*1	6	5
*MindS-bot*2	3	2
*MindS-bot*3	9	1

ID	Tot. Exp.	$\#_{foragers}$
*MindS-bot*4	9	4
*MindS-bot*5	3	0
*MindS-bot*6	10	4

reports the number of times each *MindS-bot* was observed to be a forager at the end of the experiments. In fact, a statistical analysis of this data[6] shows that (2a) is true with confidence 95%.

We are therefore in a situation in which either (1) or (2a) is true, which means that the allocation mechanism exploits mechanical differences of the *MindS-bots*. However, there is not enough data to determine which of the two conditions is verified.

5 Related Work

Other approaches to the issues described in this paper can be found in the literature. We list here a few ones.

Gerkey and Matarić (2003) review and compare some of the main task allocation methods used in the literature, where task allocation is intended as the problem of assigning tasks to one robot. They analyse the methods, that need inter-robot communication and are based either on a solution to the optimal assignment problem or on a market/auction schema, in terms of the complexity of the computation required and of the costs of communication.

The threshold-model (Bonabeau et al., 1996) is widely used in bio-inspired robotics. For instance, Agassounon and Martinoli (2002) use it for a puck-clustering problem as a means to find the optimal number of robots. Krieger and Billeter (2000) use it in a retrieval task and analyse how the performance of the group changes when increasing the group size or when communication is used.

Other works in retrieval tend to focus on the reduction of the interferences by using communication and co-ordination (Balch and Arkin, 1994), by coding territoriality in the control systems of the robots (Schneider-Fontán and Matarić, 1996), or by using heterogeneous groups (Balch, 1999).

---

[6] χ^2 test with the null hypothesis that (2b) is true and (2a) as alternative hypothesis. This test can be used only if the data sets are independent, which is granted by assuming that (1) is false.

6 Conclusions

We showed that a simple adaptation during the life time of an individual can lead to self-organised task allocation in the colony. Individuals that are mechanically better for retrieval are more likely to be selected. Future work will try to understand better these phenomena, especially in those cases in which the colony has to deal with changing environments.

Our work is also relevant for biologists. Usually, division of labour in animal colonies is explained by looking at the dimorphism of individuals, at class segregation, or also at genetic differences. However, some biologists claim that adaptation, or learning, plays an important role too, but their arguments are usually only theoretical. Our work can therefore be used to give more concreteness to their theories by using real objects in a real environment.

Acknowledgements

This work was supported by the “SWARM-BOTS Project”, funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission, under grant IST-2000-31010. The information provided is the sole responsibility of the authors and does not reflect the Community’s opinion. The Community is not responsible for any use that might be made of data appearing in this publication. Marco Dorigo acknowledges support from the Belgian FNRS, of which he is a Senior Research Associate, through the grant “Virtual Swarm-bots”, contract no. 9.4515.03, and from the “ANTS” project, an “Action de Recherche Concertée” funded by the Scientific Research Directorate of the French Community of Belgium. Jean-Louis Deneubourg acknowledges support from the Belgian FNRS, of which he is a Research Associate.

References

W. Agassounon and A. Martinoli. Efficiency and robustness of threshold-based distributed allocation algorithms in multi-agent systems. In C. Castelfranchi and W.L. Johnson, editors, *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 1090–1097. ACM Press, New York, NY, USA, 2002.

T. Balch. The impact of diversity on performance in multi-robot foraging. In O. Etzioni, J.P. Müller, and J.M. Bradshaw, editors, *Proceedings of the Third International Conference on Autonomous Agents (Agents’99)*, pages 92–99. ACM Press, New York, NY, USA, 1999.

T. Balch and R.C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1):27–52, 1994.

E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press, New York, USA, 1999.

E. Bonabeau, G. Theraulaz, and J.-L. Deneubourg. Quantitative study of the fixed threshold model for the regulation of division of labor in insect societies. *Proceedings of the Royal Society of London, Series B-Biological Sciences*, 263:1565–1569, 1996.

S. Camazine, J.-L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organisation in Biological Systems.* Princeton University Press, Princeton, NJ, USA, 2001.

Y.U. Cao, A.S. Fukunaga, and A.B. Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–27, 1997.

J.-L. Deneubourg, S. Goss, J.M. Pasteels, D. Fresneau, and J.-P. Lachaud. Self-organization mechanisms in ant societies (II): Learning in foraging and division of labor. In J.M. Pasteels and J.-L. Deneubourg, editors, *From Individual to Collective Behavior in Social Insects*, volume 54 of *Experientia Supplementum*, pages 177–196. Birkhäuser Verlag, Basel, Switzerland, 1987.

M. Dorigo, E. Bonabeau, and G. Theraulaz. Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16(8):851–871, 2000.

B.P. Gerkey and M.J. Matarić. A framework for studying multi-robot task allocation. In A.C. Schultz, L.E. Parker, and F.E. Schneider, editors, *Multi-Robot Systems*, pages 15–26. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003.

D. Goldberg and M.J. Matarić. Interference as a tool for designing and evaluating multi-robot controllers. In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-97)*, pages 637–642. MIT Press, Cambridge, MA, USA, 1997.

P. P. Grassé. La reconstruction du nid et les coordinations inter-individuelles chez *Bellicositermes natalensis* et *Cubitermes*. La théorie de la stigmergie: essai d'interpretation des termites constructeurs. *Insectes Sociaux*, 6:41–83, 1959.

M.J.B. Krieger and J.-B. Billeter. The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. *Robotics and Autonomous Systems*, 30(1-2):65–84, 2000.

T.H. Labella, M. Dorigo, and J.-L. Deneubourg. Efficiency and task allocation in prey retrieval. In A.J. Ijspeert, D. Mange, M. Murata, and S. Nishio, editors, *Proceedings of the First International Workshop on Biologically Inspired Approaches to Advanced Information Technology (Bio-ADIT2004)*, Lecture Notes in Computer Science, pages 32–47. Springer Verlag, Heidelberg, Germany, 2004.

M. Schneider-Fontán and M.J. Matarić. A study of territoriality: The role of critical mass in adaptive task division. In P. Maes, M.J. Matarić, J.-A. Meyer, J. Pollack, and S.W. Wilson, editors, *From Animals to Animats 4, Fourth International Conference on Simulation of Adaptive Behavior (SAB-96)*, pages 553–561. MIT Press/Bradford Books, Cambridge, MA, USA, 1996.