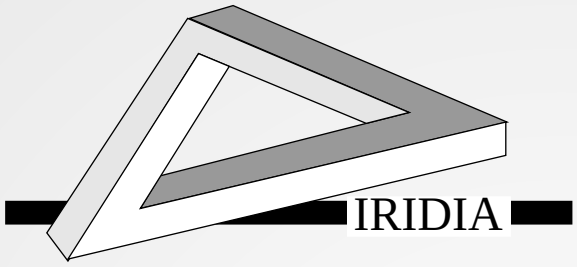

Swarm-intelligent Robotics in Prey Retrieval Tasks

Roderich GROSS

Technical Report No.

TR/IRIDIA/2003-27
September 2003

DEA thesis

Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle

50 av. F. Roosevelt • CP 194/6 • B-1050 Bruxelles • Belgium
Phone: +32-2-650.27.29 • Fax: +32-2-650.27.15

Swarm-intelligent Robotics in Prey Retrieval Tasks

by

Roderich Groß

Université Libre de Bruxelles, IRIDIA
Avenue Franklin Roosevelt 50, CP 194/6, 1050 Brussels, Belgium
rgross@ulb.ac.be

Supervised by

Marco Dorigo, Ph.D.

Maître de Recherches du FNRS
Université Libre de Bruxelles, IRIDIA
Avenue Franklin Roosevelt 50, CP 194/6, 1050 Brussels, Belgium
mdorigo@ulb.ac.be

August, 2003

A thesis submitted in partial fulfillment of the requirements of the *Université Libre de Bruxelles, Faculté de Sciences Appliquées* for the

DIPLOME D'ETUDES APPROFONDIES (DEA)

Abstract

In this work a swarm-robotic system, called *swarm-bot*, is introduced. A swarm-bot is a self-assembling and self-organizing artifact composed of a *swarm* of simple autonomous mobile robots, called *s-bots* that are able to establish (and to release) physical connections between each other and to other objects.

The swarm-bot approach emphasizes aspects like decentralized control architectures, self-assembling abilities, locality and simplicity of interactions among s-bots and between s-bots and the environment, emergence of order, flexibility and robustness.

This work addresses the problem of controlling swarm-bots or a swarm of simple s-bots in prey retrieval and collective transport tasks. We focus on tasks concerning the transport of a single prey by a group of s-bots. To accomplish the tasks efficiently, the cooperation of several group members is required.

Methods from the field of Evolutionary Computation are applied and studied to design control policies solving the underlying tasks.

Presently, experiments are carried out using a 3D physics simulator. The simulated s-bot model is an approximation of a real s-bot, currently manufactured in the scope of the SWARM-BOTS project. Swarms and swarm-bots composed of real s-bots are planned to be studied as soon as they are available.

Contents

Report Layout

This work is organized into three main chapters.

Chapter 1 provides a general introduction to related research fields. It is split into four parts. The first part provides an introduction to Evolutionary Algorithms with special focus on Evolutionary Strategies, Genetic Algorithms and Genetic Programming. The second part is dedicated to a brief introduction to *self-organizing systems*. The third part deals with distributed mobile robotic systems in general, and with the swarm robotics and the swarm-bots approach in particular. Finally, cooperative transport and retrieval strategies are discussed in the scope of (natural) ant and robotic systems.

Chapter 2 and Chapter 3 focus on particular cooperative transport problems. In each chapter the simulation framework, the experimental setup, and the results are presented.

Chapter 2 addresses the problem of the cooperative transport of a prey by a group of two s-bots into an arbitrary direction. The communication structure among s-bots is strictly limited to *interactions via environment*. Therefore, the s-bots are not able to sense each other. The task requires the cooperation of both s-bots to be completed.

Chapter 3 addresses the problem of the cooperative transport of a prey by a group of four s-bots into a specified direction. Here, *interactions via sensing* are possible. Prey objects of various mass, shape and size are utilized.

Chapter 1.

Introduction

1.1. Evolutionary Algorithms

The field of Evolutionary Algorithms does unite several fairly independently created and developed research branches started in the 60th. Certainly the most influencing ones are Evolutionary Strategies (Rechenberg 1965; Rechenberg 1973; Schwefel 1975) introduced by Ingo Rechenberg and Hans-Paul Schwefel, Genetic Algorithms (Holland 1962; Holland 1975; Goldberg 1989), founded by John Holland, and Evolutionary Programming (Fogel 1962; Fogel et al. 1966), proposed by Lawrence J. Fogel, Alvin J. Owens, and Michael J. Walsh. Within the field of Genetic Algorithms, the sub-branch of Genetic Programming (Cramer 1985; Koza 1992; Banzhaf et al. 1998) has been invented by Nichael Cramer.

The development of these branches has started in different contexts: while Evolutionary Strategies have been introduced as an all-purpose technique in experimental optimization, Genetic Algorithms have been proposed to study mechanisms of adaptive systems and to model classification processes. Evolutionary Programming has been founded to address time series problems with finite state machines created by artificial evolution.

However, these branches share similarities, such as the basic inspiration by natural evolution. Therefore, the generic term *Evolutionary Algorithms* has been established to emphasize this common base.

The following sections provides a brief summary of the development of the theory of evolution, the *Basic Evolutionary Algorithm* is described, followed by an introduction to Evolutionary Strategies.

1.1.1. Theory of Evolution

Until modern times, the belief of constancy of species - the division of living things into species would exist unchanged since time immemorial - was prevalent. The common opinion was that the diversity of nature could be reduced to a limited number of sharply defined natural types, each one defining a class of identical, constant

members.[1]

Jean Baptiste Lamarck (1744-1829) realized that species are subject of a gradually developing process. He believed in inheritance of acquired characteristics that would change according to a teleological drive towards greater perfection triggered by desires or as a result of the behavior that is influenced by those desires. In his major work "Philosophie zoologique" (Lamarck 1809) he states that frequently used organs would develop further while rarely used organs would recede.

Half a century later, Charles Darwin published his famous book "On the origin of species by means of natural selection" (Darwin 1859). Also Darwin believed living things would develop further and changes would occur mostly in small steps rather discontinuously. In fact, Darwin realized that, although development happens gradually, all living things descend from a single root. This hypothesis has been supported by the discovery of the universal genetic code of all living entities.

In opposition to Lamarck, he stated that the steps were not determined by a drive towards greater perfection during life-time, but were the result of *natural selection* - the selection of individuals being adapted best to their environment. He assumed that there would be an excessive amount of offspring, but only a limited amount of available resources in the environment. The offspring would be similar to their parents, but also vary slightly from each other. Individuals that are adapted the best to the environment would be more likely to produce offspring than those that are less adapted. This is referred by the familiar term "survival of the fittest". The continuous interplay of variation and natural selection leads to an evolutionary process.

Although Darwin assumed that characteristics would be inherited, he could not explain the underlying mechanism. The reinvention of the ideas of Mendel (Mendel 1866) at the beginning of the 20th century seemed to be incompatible with Darwin's Theory.[2] Several critics of Darwin's Theory of Evolution ("Darwinism") stated that complex organisms could arise only by macro mutations rather than a slow and continuous evolutionary process that develops gradually.

Also the key role of natural selection as one of the evolution factors was not accepted by several critics. Instead of this, neo Lamarckian, mutationist, or orthogenetic theories had been favored. However, insights, especially in microbiology, genetics, paleontology, and embryology have led to a falsification of almost all of those theories or to a support of the theory of natural selection.

Based on Darwin's Theory of Evolution and the genetic principles primarily observed by Gregor Mendel, the widely accepted Synthetic Theory of Evolution (Dobzhansky 1937; Mayr 1942) including gained insight from research in biogeography, pale-

[1] This is the belief of Essentialism.

[2] For instance Darwin's belief of continuity in the evolutionary progress and the strong discontinuity concerning inheritance of characteristics observed by Mendel.

ontology, systematics, and later also in molecular biology has been developed by Theodosius Dobzhansky and Ernst Mayr.

The synthetic theory refuses any inheritance of acquired characteristics and emphasizes the step-wise nature of evolution. It recognizes that evolutionary phenomena can always be considered as population phenomena and confirms the preeminent importance of natural selection.[3]

1.1.2. Basic Evolutionary Algorithm

The insight that natural evolution produces new organisms that are better and better adapted to their highly complex environments, has led to the application of the underlying mechanisms in different domains. For instance, Evolutionary Algorithms are applied to a huge variety of problems in optimization.

In the following, it is assumed that an optimization problem is given. A set of (feasible) *solutions* - the *search space* - is defined, each one can be assigned a *fitness* reflecting the quality of solution. The aim is to find a solution of *very high* quality.

In the context of Evolutionary Algorithms the term *individual* is used to refer to a solution. An individual is represented by its *genotype*. The genetic material that is coded in the genotype is subject to genetic operators during reproduction, but the genotype itself is immutable, so there are no changes during the lifetime. Depending on the problem domain and the evolutionary techniques used, individuals can be represented in different ways.

The genotype does contain the information to construct an organism, the *phenotype*, i.e., the expression of the properties that are coded by the genotype. The genotype-phenotype mapping can be influenced by stochastic processes. Only the phenotype is subject of selection.

The Basic Evolutionary Algorithm is illustrated in Figure 1.1. In general, Evolutionary Algorithms investigate different search paths at once. Therefore, a *population* comprised of several individuals is kept. Usually, a population of unbiased randomly initialized individuals serves as starting point. But also individuals of former evolutions or knowledge in the problem domain can be exploited in order to construct a population to start with.

The quality of each solution is determined by an evaluation procedure. Therefore, a fitness function, complex simulations, or practical experiments might be utilized in order to obtain the fitness value reflecting the solution's quality.

In case a solution of sufficient quality has been found, the algorithm terminates. Otherwise, certain individuals are selected in order to produce the population of the next generation. Thereby solutions of higher fitness are more likely to be chosen. The genetic material of the selected individuals will be modified by genetic operators

[3] Translated from (Mayr 1984).

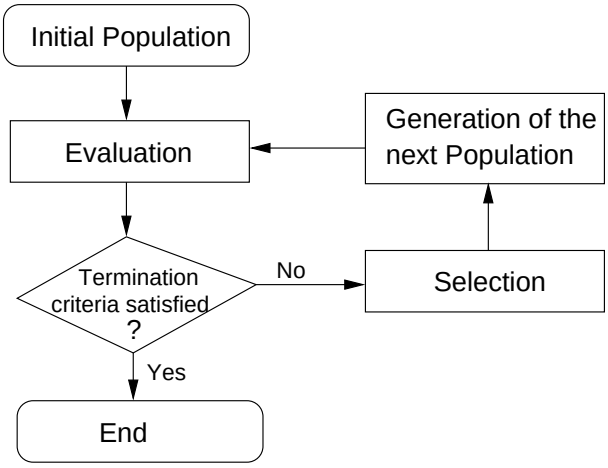

Figure 1.1.: Basic Evolutionary Algorithm

like mutation or recombination in order to produce offspring that are admitted to the population of the next generation. Mutation modifies the genotype of one single individual, whereas recombination combines the genotypes of two or more individuals. Besides this, some individuals might have the chance to replicate themselves without any difference in the genotype.

In the following sections, a brief introduction to Evolutionary Strategies, Genetic Algorithms and Genetic Programming is given. Characteristics in the individual's representation, selection, and the genetic operators are detailed.

1.1.3. Evolutionary Strategies (ES)

Representation

Problem solutions are represented as N-dimensional, real-numbered vectors, also referred to as the *object parameters*. Often, there is no encoding step needed, since a real-valued optimization problem might be given. In this case, the genotype is identical to the phenotype. The fitness can be determined using an N-dimensional objective function, that is supposed to be minimized or maximized.

Selection

Considering selection, a distinction is drawn between the so-called *plus-* and *comma-strategies*.

In case of a $(\mu+\lambda)$-ES (a plus-strategy), the best μ individuals are deterministically chosen from a set of λ offspring and μ parents. The selected μ individuals create a new set of λ offspring. Therefore, the selected set of μ reproducing individuals might

contain both - individuals from the set of λ offsprings and individuals of the former set of μ parents. This strategy uses strict elitism to preserve already evolved solutions. If and only if, at least μ better solutions have been created, an individual is replaced.

In case of a (μ,λ)-ES (comma-strategy) only the μ best solutions out of a set of λ offspring that have been created, are selected to produce offspring. Since parent individuals are not considered, there is no elitism. For this strategy λ should be bigger than μ.

Mutation

Mutation is a variation operator that changes the genome's information of a parent individual according to a suitable probability distribution. In Evolutionary Strategies mutation is usually performed by adding a vector of random components generated from a normal distribution.

One property of an Evolutionary Strategy is self-adaptive behavior. Using specific types of variation operators, in particular the mutation operator, the Evolutionary Algorithm is able to self-adapt to the local topology of the *fitness landscape*[4]. Self-adaptation can lead to a significant increase in *performance*. Performance measures have been defined in several ways (Beyer 1994). Definitions can refer to local progress in fitness landscape or search space, or global progress in search space.

A simple way to realize an adaptive mutation operator, is to control the mutation strength s. Let X be the parent's genotype and Z an N-dimensional real-valued vector of independently random components generated from a normal distribution $N(0,1)$. An offspring X' can be obtained by $X' = X + sZ$. It is pursued to control s in order to maximize performance. A rule of thumb is given by the 1/5 success rule (Rechenberg 1973): Let P_{success} be an estimation of the probability that the offspring's fitness is higher than the one of the parent individual. P_{success} is measured regularly after a certain number of generations has elapsed. If $P_{\text{success}} < 1/5$, s will be decreased, if $P_{\text{success}} > 1/5$, s will be increased, otherwise s is not modified. Keeping the success rate close to 1/5, the performance exhibited by the Evolutionary Strategy can be quite optimal. This rule of thumb has been theoretically justified and studied using some idealized assumptions concerning the fitness landscape (Beyer 1993).

For some types of problems it has been shown, that using a plus-strategy combined with a constant mutation strength leads to a sub-linear convergence to the optimum, while controlling the mutation strength leads to a linear convergence to the optimum. Comma-strategies without controlling the mutation strength do not converge.

So far, it was implicit to have isotropic mutations, that means that Z is a random vector generated according to a probability distribution with a spherical-symmetrical

[4]Metaphor figuring a combination of the search space and the fitness values of each genotype as defined by the objective function. Evolution can be thought of as a hike towards the hills of this fitness landscape. The highest peaks represent the fittest solutions.

density function. However, it is not uncommon to have specific mutation strength S_i for each dimension of the object parameter space. A self-adaptation of those strategy parameters can be reached by including them in the individual's genotype (Schwefel 1974). Then, the genotype of each individual is comprised of object parameters $X = (X_1, X_2, \ldots, X_N)$ and *endogenous*[5] strategy parameters $S = (S_1, S_2, \ldots, S_N)$.

It is assumed that *good* object parameters tend to be associated with *good* strategy parameters. In case of a mutation, first strategy parameters are mutated, then object parameters are varied using the new strategy parameters.

It is common practice to *mutate* the mutation strength parameters s_i by multiplying it with a random variable ξ with an expected value close to 1. Schwefel proposed the use of a *lognormal* distribution (Schwefel 1974). This can be realized with $\xi = \exp^y$, where $y = \tau N(0,1)$ is normally distributed. τ can be seen as a learning parameter or learning rate that specifies how fast the Evolutionary Strategy is able to adapt. Schwefel's rule suggests to set $\tau \propto \frac{1}{\sqrt{N}}$ (for an analysis see (Beyer 1996)).

A refinement of this is to set $\xi = \exp^{y_g + y_s}$, where $y_g = \tau_g N(0,1)$ is calculated once per individual and therefore unique for all object parameters, while $y_s = \tau_s N(0,1)$ is determined for each object parameter again. Thereby, there is a common component for all strategy parameters and there are specific ones. τ_g can be set to $\frac{1}{\sqrt{2N}}$ and τ_s can be set to $\frac{1}{\sqrt{2\sqrt{N}}}$.

In principle, components of the mutation vector Z do not need to be independent. One might think of a matrix of strategy parameters that specifies the rotation for all pairs of axes of the N dimensional search space. However, the number of strategy parameters might get intractable for high dimensional search spaces.

Recombination

A recombination operator combines ρ parent individuals to create a new offspring ($2 \leq \rho \leq \mu$). Each individual out of the set of the best μ has the same probability to be selected. Multiple selections are possible.

The intermediate recombination operator computes the center of mass of the real-valued vectors given by the parent individuals. The dominant or discrete recombination operator randomly selects parameters from the ρ parents (for each coordinate).

Under certain assumptions it can been proven that a choice of $\rho = \mu$ leads to the best performance (Beyer 1995). In case of the intermediate recombination, one might partition each parent vector into a component directing to the global optimum, and a second component perpendicular to this. Concerning the components perpendicular to the optimum, parent individuals have no preferential direction. Building

[5] Endogenous strategy parameters are adapted during evolution, while exogenous strategy parameters (e.g. μ or λ) are kept constant.

the center of mass, the impact of those components tends to disappear. Therefore, recombination can be seen as an operator repairing defective genes (H.-G. Beyer 2001).

1.1.4. Genetic Algorithms (GA)

TODO

1.1.5. Genetic Programming (GP)

TODO

1.1.6. Evolutionary Robotics

TODO

1.2. Self-organizing Systems

The theory of self-organization has been developed by many scientific fields, including chemistry, physics, biology, cybernetics, and economics. In general, the term *Self-organizing System* denotes systems with the ability to change their internal structure and their function in response to external circumstances.

The Encyclopedia of Physical Science and Technology (Banzhaf 2002) gives a summary of the variety of features that have been identified as being typical for many self-organizing systems:

> Self-organizing systems are *dynamic*, *non-deterministic*, *open*, exist *far from equilibrium* and sometimes employ *autocatalytic amplification of fluctuations*. Often, they are characterized by *multiple time-scales* of their internal and/or external interactions, they possess a *hierarchy of structural and/or functional levels* and they are able to *react* to external input in *a variety of ways*. Many self-organizing systems are *non-teleological*, i.e. they do not have a specific purpose except their own existence. As a consequence, *self-maintenance* is as important function of many self-organizing systems. Most of these systems are *complex* and use *redundancy* to achieve resilience against external perturbation tendencies.

Numerous attempts to define the notion of *self-organization* can be found in the literature (Ashby 1947; Nicolis and Prigogine 1977; Haken 1988; Coveney and Highfield 1995). Camazine et al. define self-organization as

> a process in which pattern at the global level of a system emerges solely from numerous interactions among the lower components of the system. Moreover, the rules specifying interactions among the system's components are executed using only local information, without reference to the global pattern" (Camazine et al. 2001).

The term *pattern* can refer to a "spatio-temporal physical structure or behavior" (Anderson 2002). Patterns *emerge* by numerous interactions among lower-level components or possibly even by many repeated interactions belonging to a single individual (Anderson 2002).

In general, the term *emergence* denotes the appearance of qualitatively new phenomena on higher levels of a hierarchical system. The emergence of a pattern in a self-organized system is caused within the system and is "not generated from interference or other external guiding forces, such as templates" (Anderson 2002; Camazine et al. 2001). Nevertheless, self-organized systems can be affected by the environment. Without any change of the characteristics of underlying lower-level components, they may switch between different semi-stable states (multi-stability) due to *intrinsic factors* such as random fluctuations within the system but also due to *extrinsic factors* such as environmental changes (Deneubourg et al. 1989; Anderson 2002).

Since the rules specifying interactions among the rather simple lower-level components of the system use only local information comparatively limited cognitive abilities and knowledge of the environment (if any) are required at individual level.

Self-organized systems are usually regulated by *positive-* and *negative feedback* that holds between the system's low-level components. These components affect each other in a mutual way. Thus, circular cause-and-effect relations are present. Positive feedback corresponds to a recurrent influence that amplifies the initial state. This results in growing deviations in a runaway, explosive manner. Negative feedback stabilizes the system, by bringing deviations back to their original state, for instance if available resources are exhausted (Heylighen 2002).

Positive feedback is seen as an important "key ingredient" of many self-organized systems. In some self-organized systems such as "thermoregulation in honeybee (*Apis mellifera*)" positive feedback seems not to be present or the presence is not obvious (Anderson 2002).

In nature, self-organizing systems have been identified in living systems (cells, neural networks, flocks of birds, schools of fish, swarms of bees) and non-living systems (sand dunes, galaxies, stars). Additionally, self-organization has been discovered "in man-made systems (societies, economics)" and the "world of ideas (world views, scientific believes, norm systems)" (Banzhaf 2002).

1.3. Distributed Mobile Robotics

The field of distributed robotics has received growing attention by researchers in recent years. Systems comprised of multiple mobile robots have become subject to first studies in the late 1980's. Early investigations include works on cellular (or reconfigurable) robot systems (Fukada and Nakagawa 1987; Beni 1988), multi-robot cooperation (Asama et al. 1989) and multi-robot motion planning (Premvuti and Yuta 1990).

Multi-robot systems have been studied in various "principle topic areas" (Mataric 1995; Parker 2000) addressing biological inspiration, communication, architectures and learning, as well as in different application domains, e.g., exploration (Billard et al. 1999; Hayes et al. 2000), object manipulation (Tung and Kleinrock 1993; Parker 1994; Donald et al. 1994; Sen et al. 1994; Martinoli and Easton 2003) and foraging (Goss and Deneubourg 1992; Drogoul and Ferber 1993; Becker et al. 1994; Goldberg and Matarić 1999).

Related research previous to investigations concerning multi-robot systems has focused on the control of a single robot, or addressed the field of Distributed Artificial Intelligence[6].

In the next section potential advantages of multiple mobile robotic systems are presented. In Section 1.3.2, the terms *collective-* and *cooperative behavior* are briefly introduced. Important issues of multi-robot system architectures are detailed in Section 1.3.3.

Section 1.3.4 presents a novel approach to the design and implementation of collective robotic systems, taking inspiration from insect societies.

1.3.1. Potential Benefits of Having Multiple Mobile Robots

Systems of multiple mobile robots are of particular interest due to the following reasons (this list is not exhaustive).

Task Complexity and Solvability Multiple robots may be able to solve tasks that are inherently too complex or even impossible to be accomplished by a single robot; moreover, a robotic system can gain benefit in performance from using more than one robot:

- Multiple robots can distribute themselves in the environment to occupy several potentially distant locations at the same time, while the current operating range of a single robot is limited to the robot's vicinity.

[6]Distributed Artificial Intelligence (DAI) (Huhns 1987; Gasser and Huhn 1989; Bond and Gasser 1988) is concerned with the cooperative solution of problems by a decentralized group of *agents*. The field can be divided into two parts: Distributed Problem Solving (DPS) and Multi-agent Systems (MAS) (Durfee and Rosenschein 1994; Mataric 1995).

- Multiple robots are well suited to perform several actions in parallel.
- Task decomposition and task allocation strategies can be exploited in multi-robot systems in order to gain improved performance.

Design Costs and Versatility Multi-robot systems can be composed of simple robots with identical hardware specification. Therefore, the robots may be manufactured with comparably little cost. Moreover, a single robot that is able to accomplish a predefined task on its own might be too specialized to perform other tasks, while groups of multiple robots tend to be applicable in different problem domains.

Flexibility, Robustness and Fault-tolerance A distributed robotic system may be able to act more flexible to environmental changes than a single robot can. Moreover, multi-robot systems introduce redundancy. If entire robots or parts of them fail to serve their purpose, the whole system might still be able to complete the task. Flexibility and robustness are key properties observed in social insect colonies (see Section 1.4.1). Various works in distributed robotics draw inspiration from this field (Deneubourg et al. 1990; Kube and Zhang 1992b; Drogoul and Ferber 1993; Kube and Zhang 1995; Martinoli and Mondada 1995; Martinoli et al. 1997; Martinoli and Easton 2003).

Finally, multi-robot system are of general research interest since their development may yield insights into fundamental problems of fields spanning the social sciences (organization theory, economics) and life sciences (theoretical biology, animal ethology) (Cao et al. 1997).

1.3.2. Collective and Cooperative Behavior

The term *collective behavior* is commonly used to refer to any behavior of robots in a multi-robot system. *Cooperative behavior* is a collective behavior that is characterized by *cooperation*. Various definitions of cooperation and cooperative behavior can be found in the literature. Cooperative behavior can be defined in the following way (Cao et al. 1997):

> Given some task specified by a designer, a multi-robot system displays **cooperative behavior** if, due to some underlying mechanism (i.e., the "mechanism of cooperation"), there is an increase in the total utility of the system.

Martinoli and Mondada distinguish between *collective noncooperative* tasks, do "not necessarily need cooperation among the individuals to be solved" and *collective*

cooperative tasks, "which absolutely" need "the collaboration of two or more individuals in order to be carried out, because of some physical constraints of a single agent" (Martinoli and Mondada 1995).

1.3.3. System Architectures

In the following, important issues concerning the architecture of multi-robot systems are discussed.

Centralization/Decentralization Concerning the control, multi-robot systems are usually classified as either being centralized or decentralized. The term *centralized* denotes the characteristic that only one single control agent is present. Although there is no theoretical or empirical comparison between both approaches (to the best of our knowledge), centralized control systems are often said to be not well suited for large group sizes due to the increasing demand for communication. Moreover, the whole system will collapse, in case the central controller fails.

Homogeneous and Heterogeneous groups An important issue concerning architectures of multi-robot systems is the question of similarity between the robots of a team. Robots can be equipped with identical *capabilities* or not. The term *homogeneous* group of robots refers to the first case. If one or more robots are different from the others, the term *heterogenous* group is used. Concerning the robot's capabilities, it is common to include not only the abilities of the robot's sensors and actuators, but also its computational and behavioral abilities. Most research has been carried out with homogeneous systems.

Balch studied behavioral specialization in learning robot teams. Inspired by Shannon's information entropy (Shannon and Weaver 1949), he proposes the metric *social entropy* to evaluate diversity in societies of mechanically similar but behaviorally heterogeneous agents (Balch 1997b). Balch studies the quantitative relationship between performance and behavioral diversity on various tasks. In case of a robot soccer task behaviorally heterogeneous robot teams can lead to a performance benefit (Balch 1997a). For a particular foraging task the homogenous robot teams performed better (Balch 1999).

Heterogeneous groups of mechanically different robots have been studied with the multi-robot system ALLIANCE. ALLIANCE is a distributed control architecture for small- to medium-sized heterogeneous robot teams (Parker 1994; Parker 1999). Cooperation is achieved using broadcast communication allowing robots to be aware of the actions of their teammates. Teams can be composed of both legged and wheeled robots.

Communication Another way to categorize multi-robot systems is the communication structure used to interact. Different types of interactions can be distinguished (Cao, Fukunaga, and Kahng 1997):

- Interaction via the environment. In this case there is no explicit communication between the robots. Moreover, robots are not able to directly perceive each other. Interactions are based on modifications of the environment. This kind of indirect interaction using the environment as communication media is known as *stigmergy* (Grassé 1959). Stigmergic communication is observed, for example, in social insects like ants (see Section 1.4.1). There are numerous works using this kind of indirect communication in the context of multi-robotic systems (Goss and Deneubourg 1992; Becker et al. 1994; Arkin 1992).

- Interaction via sensing. In this case no explicit communication is prevalent, but a robot can locally sense the presence of other teammates. The term *kin recognition* refers to the ability to distinguish teammates from other objects in the environment (Mataric 1994). Kuniyoshi et al. studied the recognition of the teammates' actions to gain useful information about the current situation for the coordination of autonomous agents in various tasks (Kuniyoshi et al. 1994). They propose a framework, called *Cooperation by Observation* that is based on interactions via visual action recognition. Moreover, they introduce the term *attentive structure* that is "a set of attentional relations among all members of a cooperative group and related objects".

  Attentive structures exist in social animals like monkeys or apes: the society members are paying attention to a common leader individual. Their actions can be highly dependent on the observed behavior of the leader, for instance during an attack of the group (Chance and Jolly 1970).

  Herding, flocking, and schooling behaviors in animals are examples of attentive structure without the presence of any leader individual. Flocking, dispersion and pattern formation has been studied in the context of multi-robot systems (Mataric 1992a; Mataric 1992b; Trianni et al. 2003).

- Interaction via communication. This class of interactions refers to explicit communication among robots. Robots can broadcast messages or send messages dedicated to a particular teammate.

  Bach and Arkin showed that using communication can be beneficial for particular types of tasks. Reactive behaviors have been developed in simulation and have been ported for the purpose of validation on a group of real mobile robotics system (Balch and Arkin 1994). Already a small amount of transmitted information could lead to a significant increase in performance.

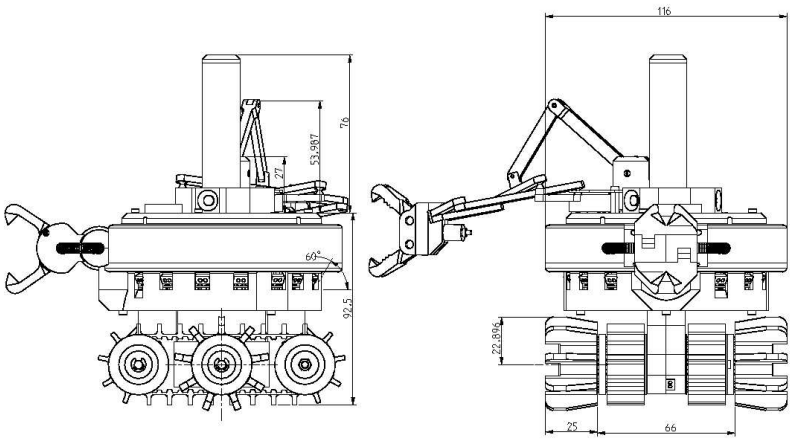

Figure 1.2.: Illustration of the s-bot prototype (side and front view): the s-bot is equipped with a *Differential Treels*© *Drive* that is a combination of tracks and wheels, a gripper fixed to the turret, a gripper attached to a flexible arm, and an omnidirectional camera system placed inside the pillar on the s-bots top. The turret can be rotated with respect to the chassis.

1.3.4. Swarm Robotics and Swarm-bots

Classical AI approaches tend to model robots as rational, deliberative agent, able to search for solutions within abstract model of their environment.

In *swarm robotics* and in the field of *Swarm Intelligence* (Bonabeau, Dorigo, and Theraulaz 1999) in general, a fundamental different view of intelligence is taken: rather than modeling robots to be rational and deliberative, individuals are of low complexity having limited cognitive and computational abilities. In fact, complexity on the level of a collective of simple individuals can emerge due to numerous interactions among the system's components. This is a well-known property of a self-organizing system (see Section 1.2). Examples can be found in nature: social insects – ants, bees, termites and wasps – can be viewed as powerful problem-solving systems with sophisticated collective intelligence.

Swarm robotics is a novel approach to the design and implementation of collective robotic systems, taking inspiration from insect societies. The swarm robotics approach emphasizes aspects like decentralized control architectures, locality and simplicity of interactions among robots and between robots and the environment, emergence of order, flexibility, miniaturization, and robustness. Swarm robotic systems are composed of *swarms* of robots that tightly interact and cooperate to reach their goal.

The study of the design, hardware implementation, test and use of self-assembling swarm robotic systems called *swarm-bots* is the objective of the SWARM-BOTS

project[7] (Mondada et al. 2002; Mondada et al. 2002; Şahin et al. 2002). A *swarm-bot* is a self-assembling and self-organizing artifact composed of simple autonomous mobile robots (called *s-bots*) that are able to establish and to release physical connections between each other.

The real swarm-bot will be composed of a number of rather simple, insect-like, s-bots, built out of relatively cheap components. An early prototype of the real s-bot is illustrated in Figure 1.2. The s-bot is equipped with a *Differential Treels*© *Drive* that is a combination of tracks and wheels. The gripper fixed to a turret can be used to establish robust connections to other s-bots or objects. Additionally, a gripper attached to a flexible arm can be utilized to grasp other s-bots or objects. An omnidirectional camera system is placed inside the pillar on the s-bots top. The turret can be rotated with respect to the chassis.

1.4. Cooperative Prey Retrieval

1.4.1. Cooperative Prey Retrieval in Ants

This section gives a brief introduction about prey retrieval in ant colonies.

Let us consider a foraging ant that discovers a prey item to retrieve in an environment. The ant will first try to carry the item. If it turns out that the item is too heavy, the ant will drag it. If the item resists motion, the ant will realign its body without releasing the grasped item to try to drag again. If several attempts to drag the item fail, the ant releases the item to find a better position to grasp. If all attempts are without success, the ant might give up or recruit other nest-mates.

Two types of recruitment strategies can be distinguished: in short-range recruitment, an ant emits a chemical signal in the air to attract other nest-mates in the immediate vicinity. In long-range recruitment, an ant returns back to the nest laying a pheromone trail. The pheromone trails guide nest-mates to the prey. It has been observed that both short-range and long-range recruitment techniques appear in the species *Novomessor cockerelli* (Hölldobler and Wilson 1990).

A group of ants might cut the prey into small pieces that can be retrieved in solitary transport. However, many species of ants can retrieve heavy items cooperatively (Sudd 1963; Robson and Traniello 1998; Traniello and Beshers 1991).

In case the prey that is retrieved starts to resist motion after some time, realigning and repositioning behaviors appear that resemble the ones described for single ants discovering new prey items (Sudd 1965). In this way, deadlock and stagnation situations can be solved.

Group retrieval can be much more efficient than the alternative of cutting the item

[7] SWARM-BOTS is a project sponsored by the Future and Emerging Technologies program of the European Community (IST-2000-31010).

into small pieces and retrieving the pieces by single ants. It has been reported that a group of 100 workers of the *Pheidologeton diversus* can transport an item weighting 5000 times as much as a single worker, and the weight carried by workers involved in solitary transport on the same foraging trail was at most 5 times the individual weight (Moffet 1988; Bonabeau et al. 1999).

The size of the group of workers engaged in the transport of a prey is adapting to the degree of difficulty encountered while trying to retrieve the prey (Hölldobler 1983). This is achieved merely by the recruitment strategies that cease to attract other nest-mates in case the prey can already be moved with the available amount of workers.

It has been observed that ants do not interact directly but indirectly: The behaviors of ants modify the environment and thus affect the behaviors of other ants. This is referred as the concept of stigmergy (Grassé 1959). The *indirectness* of interactions is evident for the recruitment strategies used. Another example is that ants engaged in group transport coordinate via the item being carried.

Although single ants exhibit comparatively simple behaviors on the individual level, the colony or even parts of the colony are able to solve complex tasks. The emergence of complexity at a global level in a system composed of simple agents with numerous local interactions and without reference to the global pattern can be explained by the concept of self-organization (see Section 1.2).

1.4.2. Cooperative Transport by Robotic Systems

Deneubourg et al. proposed the use of self-organized approaches for the collection and transport of objects in unpredictable environments (Deneubourg et al. 1990). Each robot unit could be simple and inefficient by itself, but the group of robots would have a complex and very efficient behavior generated by the interactions between robots. Cooperation could be achieved without any direct communication among robots, as in some biological systems that rely on indirect communication through the environment or that exhibit particular individual behaviors (Grassé 1959; Deneubourg and Goss 1989). In a transport task, for instance, coordination could occur by inter-individual competition and via the object to move.

Kube and Zhang studied a decentralized approach for a group of simple robots to push an object that cannot be moved by a single robot (Kube and Zhang 1992a; Kube and Zhang 1992b; Kube and Zhang 1995). The approach does not make use of explicit communication among robots. Taking inspiration from biological collective transport systems like ant colonies, they extended the model by a stagnation recovery mechanism. Moreover, a "first formalized model of cooperative transport in ants" has been provided (Kube and Bonabeau 2000). A 2D simulator and later on real robotic systems have been used for validation.

One possible drawback of the control policies described by Kube and Zhang seems

to be the lack of efficiency concerning the time required to accomplish a transport task, and the rather high number of robots engaged in the transport of an object that could already be moved by a group of two robots.

Other approaches rely merely on global planning or on a combination of local and global planning with centralized conflict resolution (a list is given in (Bonabeau et al. 1999)). The work of M.J. Mataric, M. Nilsson, and K.T. Simsarian on groups of two cooperative box-pushing robots can be included to the list, since all robots share all sensory inputs, even if they might respond autonomous (Mataric et al. 1995).

Lynne Parker studied the use of distributed control systems in heterogeneous groups of two robots (Parker 1994; Parker 1998; Parker 1999) on certain tasks including box-pushing. One robot had two wheels arranged as a differential pair. The other one was a robot with six legs. Parker developed a learning system in which robots are able to learn from their previous experience with other robots. Investigations were made both with simulated and with physical mobile robot teams.

Chapter 2.

Evolving Cooperative Transport Strategies for two minimalistic s-bots

Generally, we are interested in the design of policies controlling a whole colony or just a group of simple s-bots engaged in prey retrieval tasks. Desired properties concerning the behaviors to be obtained on the colony or group level include:

- Efficiency: the *more* prey are retrieved by the colony or group of s-bots, the better.[1] *Clever* search and recruitment strategies to find and to select the items to retrieve as well as fast transport strategies are of particular interest. To increase efficiency, strategies might exploit observable and predictable characteristics of heterogeneous distributions of prey in time and/or space, e.g. by making use of learning techniques in order to let the colony, group, or individual forager adapt to current or coming environmental conditions.

- Robustness: the colony or group of s-bots should be able to accomplish the task even though entire s-bots or s-bot parts may fail to serve their purpose. Sensor and actuator devices of same type behave not identical, sensors provide uncertain values, and the physical responses concerning commands sent to actuators are not precisely predictable. Nevertheless, successful strategies should be able to accomplish the task with moderate decrease of performance.

- Flexibility: the colony or group of s-bots should be able to adapt to changing environments and various initial conditions. Flexibility is desirable on both, colony and group level. For instance, on the colony level, the system should be able to adapt to the current distribution of prey items in the environment in order to exploit its characteristics (e.g. huge clusters of prey items) to perform efficiently. On group level, the s-bots should be able to form suitable structures in order to retrieve prey of various shape and size. Once the prey is moving, the group should be able to avoid stagnation in case obstacles are encountered.

[1] Alternatively, a definition of efficiency can additionally take the amount of energy spent by the s-bots into account.

- Scalability: it is desirable that the obtained strategies can be successfully applied to larger group sizes or bigger colonies. Having a larger number of s-bots, it is possible to retrieve *more* prey in parallel. Moreover, prey objects having a higher resistance to motion can be transported.

In our experiments, the colony or the groups of s-bots will be controlled by decentralized control systems. Principally, each s-bot is a fully autonomous, mobile robot. Centralized control systems are said to be not well suited for large group sizes due to the increasing demand for communication. Moreover, the whole system will collapse, in case the central controller fails (see also Section 1.3.3).

In the design process of our multi-robot system, we focus on mechanically similar groups of s-bots. In order to control the s-bots they are initially equipped with identical (artificial) recurrent neural networks. Since, recurrent neural networks are able to adapt their behaviors during life-time, some s-bots of the group might exhibit behaviors that are very different from the ones of the other group members. In this sense, the group of s-bots is heterogeneous (see also Section 1.3.3).

In this chapter we apply an artificial evolution to synthesize neural networks in order to control a small group of simple s-bots engaged in a cooperative transport task. We consider the case that only one single prey is present. Furthermore, no advanced search or recruitment techniques are necessary, since all s-bots are placed in the immediate vicinity of the prey.

The environment is a flat ground of infinite size. Since navigation on rough terrain is a challenging research topic for its own, it would probably increase the problem complexity to a high degree. Investigations concerning prey retrieval on rough terrain will be carried out in the future.

The environment cannot be perceived precisely since sensors deliver uncertain values. Commands sent to the s-bots' actuators result in an imprecise, delayed response. Moreover, sensors and actuators of same type behave not identical.

There is no explicit communication among s-bots. Moreover, the s-bots are not able to sense each other. The communication structure is limited to *interactions via the environment* (see Section 1.3.3).

The following two sections introduce the underlying simulation framework and the task that is studied. In Section 2.3 the experimental setup is detailed. In Section 2.4 and Section 2.5 the results and conclusions are presented.

2.1. Simulation Framework

To study multiple s-bots pulling and pushing items of prey in simulation, it is essential to consider the physical characteristics of interactions among agents as well as between agents and their environment. Therefore, the level of abstraction in the

Table 2.1.: Friction coefficients belonging to pairs of material types in contact. The "-"-symbol indicates that no contact of the corresponding pair of material types will occur in the experiments described in this chapter.

	ground	wheels	torso	prey
ground	-	0.8	-	0.25
wheels	0.8	-	-	-
torso	-	-	0.02	0.02
prey	0.25	-	0.02	-

model is chosen to be sufficiently low. In order to have a somewhat acceptable simulation speed, a numerical solver is used to perform the calculations concerning the collisions among objects and the body dynamics.[2]

The simulated s-bot model is an approximation of a real s-bot, currently manufactured within the SWARM-BOTS project. A projects description and an illustration of an early prototype of the real s-bot can be found in Section 1.3.4.

In general, simulation models have to be defined carefully, in case it is intended to port control algorithms from simulated to real robots. Since we have not yet gained much experience concerning the real s-bot prototype, we do not expect that control algorithms let the s-bots behave in a simular way both in simulation and on the real s-bots. However, focus is given on the application of techniques of *Automatic Programming* that, in principle, can be applied directly on real s-bots. Finally, the gained insight in the mechanism underlying simulated prey retrieval systems may lead to a better understanding of the tasks in general.

The following two sections are dedicated to the description of the s-bots environment and the s-bot itself. The cgs metric system is used, so lengths are measured in cm, masses are measured in grams, and time is measured in seconds.

2.1.1. The Environment

In the experiments described in this work, a flat ground of infinite size is present. The gravity is set to $981\,\mathrm{cm\,s^{-2}}$.

In order to simulate reasonable frictional forces, different material types are used to model the ground, the s-bot's wheels, the s-bot's torso and the prey (the s-bot is illustrated in Figure 2.1). To calculate the frictional forces, an approximation based on the Coulomb friction law is used. The friction coefficients concerning possible contacts among material types are presented in Table 2.1.

[2] The core simulator system has been developed in cooperation with IDSIA (*Dalle Molle Institute of Artificial Intelligence Studies*), Switzerland. It is based on libraries of the commercial physics engine Vortex™ from CMLabs.

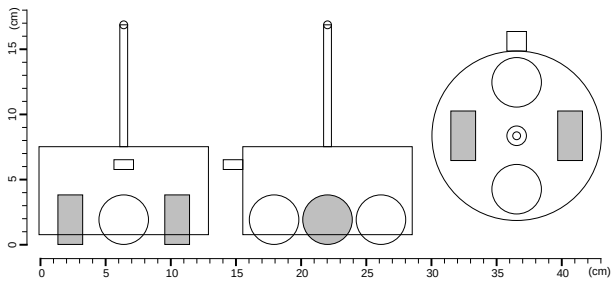

Figure 2.1.: Front, side and top view of the s-bot: the cylindrical torso is equipped with a small gripper element heading to the front, and a camera placed on the pillar. Two passive, spherical wheels (white) are connected to the torso's back and front part. Two motorized, cylindrical wheels (grey) are connected to the torso on the left and right side. Collisions between wheels and the torso are disabled.

The friction coefficient between the s-bot's wheels and the ground is set to 0.8. However, s-bots can still turn easily, and s-bots can pull or push each other, since a slip between the wheels and the ground is defined.

The friction coefficient between the prey item and the ground is set to 0.25. This value is low enough to avoid that a prey item starts jolting when pulled or pushed.

Prey are modeled as cylinders. Mass, radius and height are specified by each experimental setup.

2.1.2. The S-Bot

The s-bot model is shown in Figure 2.1. The body is composed of a cylindrical torso, a *gripper* element that is fixed on the torso's edge heading to the s-bot's front, a long cylindrical pillar placed on the top of the torso to support a visual camera system, and four wheels: two active wheels on the left and right, and two passive ones in the front and the back. The torso has a diameter of 13 cm and the whole s-bot has a height of 16.85 cm. However, the center of mass is below a height of 4 cm, since the torso element has a height of only 6.75 cm and it's placed close to the ground (see Figure 2.1).

The passive wheels are modeled using sphere geometries of radius 1.9 cm connected to the torso via a ball-and-socket joint. The active wheels are modeled by cylinders of radius 1.9 cm and height 1.9 cm connected to the torso via a motorized hinge joint. They are not equipped with a steering system.

The mass of each wheel is 20 g. The whole s-bot has a mass $m = 660\,\mathrm{g}$. Since gravity is set to $981\,\mathrm{cm\,s^{-2}}$ the s-bot's weight, i.e., the force that acts on the mass due to gravity is equal to $660\,\mathrm{g} \cdot 981\,\mathrm{cm\,s^{-2}} = 647{,}460\,\mathrm{dyne}$.[3]

[3] 100,000 dyne = 1 Newton.

The control system reads the sensory status and sets actuator activations every 0.1 seconds. This is equal to the control frequency that can be used by certain real, mobile robots, for instance the Khepera Robots (Mondada et al. 1993). Too high frequencies are not reasonable, due to several reasons. First, reading a sensor value of a physical or chemical sensor needs a certain amount of time. Second, after setting the activation of an actuator, there will not be an immediate response. And third, the s-bot has only a limited amount of computational resources. In case of the camera sensor for example, feature extraction algorithms will take some time to be accomplished.

In the following two sections the actuators and sensors of the s-bots are detailed.

Actuators

The s-bot is equipped with two active wheels and a gripper element.

The active wheels are connected to the torso via a motorized hinge joint. The maximum torque that can act on an active wheel has been chosen to be 200,000 dyne cm, corresponding to a force (applied at a distance of 1 cm) that is a little bit less than $\frac{1}{3}$ of the s-bot's weight. An active wheel can be controlled by setting a desired angular speed that is pursued respecting the given torque limit. In case the desired speed is negative the wheel turns backward, otherwise it turns forward. In both directions, the angular speed is restricted by the upper bound $15\,\mathrm{s}^{-1}$. Note that the wheel may slide in case the speed is to high.

To simulate an inaccurate behavior like it is known from real actuators, the desired speed s for a wheel is changed according to the following algorithm:

$$\begin{aligned}&\xi_{r1} = \mathrm{GAUSS}(1, 0.05);\\ &s = \xi_{r1} \cdot \xi_{r2} \cdot s;\\ &\text{if } \ (|s| < \xi_t)\ s = 0;\end{aligned}$$

where $\mathrm{GAUSS}(\mu, \sigma)$ provides a random value following the distribution $N(\mu, \sigma)$, i.e. the normal distribution with expected value μ and standard deviation σ. ξ_{r1} is generated each time step specifying a current relative error for a particular wheel, while (ξ_{r2}, ξ_t) specifies a permanent deviant behavior for a particular wheel persisting for the whole simulation period. ξ_{r2} gives a relative error like ξ_{r1} does. ξ_t is a threshold specifying the minimum non-zero absolute value of the desired speed the wheel can be set to. ξ_{r2} and ξ_t are determined for each life-time according to the normal distribution $N(1, 0.02)$ and the uniform distribution $U(0.1, 0.5)$. In this way, some wheels are biased to turn faster than others, the wheel's speed is fluctuating, and activations less than certain threshold values will not result in any movement of the wheel.

The gripper element is a controllable, sticky box heading to the front. If the element is in *gripping* mode, a connection will be established when the gripper element is in contact with a grippable object. This is realized by dynamically creating a ball-and-socket joint connecting the s-bot's torso and the object. The joint is positioned on the gripper element. Once the gripper is set to the *open* mode, this joint will be removed to release the object.

In our experiments, both the prey and the s-bots' torsos are grippable objects.

The connection formed by the gripper element will break if too much force is transmitted via the corresponding joint. This is a characteristic of the real s-bot and of gripper elements in general. One strategy to transport a heavy item by multiple robots is the formation of a chain pulling the item. However, as long as there is no rope or something equivalent to transmit forces, the chain will break, in case too high forces do appear. To overcome this problem, strategies might form structures having multiple connections to the object and/or let the robots push the prey instead of pulling it.

The limit for the force acting on the gripper element has been chosen to be 1,000,000 dyne (i.e. 10 Newton). This limit still permits s-bots engaged in the transport of a heavy prey to form structures like small pulling chains.

Sensors

The s-bot is equipped with a camera and a gripper status sensor. These sensors have their counterparts on the real s-bot. A summary of the information provided by each sensor is presented in Table 2.2.

The camera is mounted on a pillar support that is fixed at the center of the torso's top. The camera sensor is situated 16.85 cm above the ground. In principle, it is able to sense objects in all horizontal directions having an infinite range. However, the quality of the perceived signal decreases when the object's distance is increased. Therefore, the sensing range is restricted depending on the type of feature that is supposed to be extracted.

The simulated camera provides data based not on recorded images but on information available in the simulator, e.g. the distance to another object. The implicit assumption here is that in case of the real s-bot, such information can be obtained from the camera using feature extraction algorithms.

The camera sensor can provide information about a cylindrical prey object in the surrounding, if the horizontal distance between the cylinder's border and the camera is at most $R = 50\,\text{cm}$. R can be called the sensing range. In particular, the camera sensor is able to detect the horizontal distance between the s-bot and the closest point of the prey's border, and the horizontal angle to the corresponding point with respect to the s-bot's heading.

Moreover, the camera sensor is able to detect the horizontal direction of the highest

Table 2.2.: List of information provided by the sensors. In the experiments described in this chapter, the s-bots are equipped with the gripper status sensor and the camera sensor.

Sensor device	Information provided
gripper status sensor	status of being connected to another object via the gripper element
camera sensor	horizontal angle and distance to the a cylindrical prey object (sensing range: 50 cm)
	horizonatal angle of highest intensity of light perceived (a constant light signal is emitted by a beacon)

intensity of light perceived. Light is emitted by a single beacon that is placed in the environment. The beacon distributes a light signal of infinite range. It can guide the s-bots involved in the transport task to indicate the direction, the prey is desired to be moved. In practice, this kind of *global* information might not be available due to environmental constraints like walls or a rough terrain.

The beacon might be replaced by systems where signals indicating the desired direction of movement are emitted by specialized s-bots instead. These s-bots could distribute themselves in the environment, to establish a path from the prey to the target position, e.g., a nest. Such a chain of specialized light emitting s-bots could guide the group of s-bots involved in the transport task from the current position to the nest.

In case of real s-bots, the camera sensor will be affected by noise. Therefore, noise is modeled in the simulation too. In case of the light or the prey perceived, a angular deviation r_α is added to the horizontal angle α_{rad} (radian measure) that indicates the target direction to the light or the prey. r_α is generated using a normal distribution with expected value 0 and standard deviation 0.1. Also the measured distance (in cm) to a perceived prey object is affected by noise. This is modeled by adding a random variable following a normal distribution $N(0, 1)$.

A gripper status sensor is able to perceive whether the gripper element is connected to another object or not.

2.2. The Task

The Objective Our aim is to control a group composed of 2 simple s-bots to cooperate to transport as far as possible a heavy prey of specific shape, size and mass within a fixed time period into an arbitrary direction. The prey cannot be moved without cooperative behavior.

The Environment The s-bots' environment consists of a flat ground. A cylindrical prey of radius 12 cm, height 10 cm and mass 500 g is placed in the center. A light emitting beacon is placed 300 cm away. The s-bots are placed at random positions and orientations, but not more than 25 cm away from the edge of the cylindrical prey.

The S-bot's Capabilities The s-bots can localize the prey object and they can sense the beacon that is placed in the environment. The s-bots are able to control their gripper elements and they perceive whether a connection is established or not.

There is no explicit communication between s-bots. Moreover, s-bots are not equipped with any sensor to perceive their teammates. However, due to the simulated physical embodiments of s-bots, they are interacting with an environment that is in turn changed by their action.

The s-bots sensors and actuators are affected by various types of noise. Moreover, sensor and actuator devices of same type behave not identical.

2.3. Experimental Setup

2.3.1. The Neural Network Controller

Each s-bot of a group acting in a simulation environment is initially equipped with an identical neural network. However, as recurrent neural networks are not restricted to reactive behaviors, each individual s-bot can change its state during life-time according to the sensory input it has perceived. Therefore, it is possible that s-bots of the same group behave completely different, even if placed in the same environmental context.

To control the s-bots, an Elman Network (Elman 1990) is used. Elman networks are recurrent neural networks having a fully interconnected hidden layer of neurons, as can be seen in Figure 2.2. The set of neurons N of the Elman network can be partitioned according to four different types of neurons: the bias neuron b, the set of input neurons I, the set of hidden neurons H, and the set of output neurons O. The set C of connections among neurons is given by

$$\begin{aligned} C =& \{(i,j) | i \in \{b\} \cup I \cup H, j \in H\} \\ & \cup \{(i,j) | i \in \{b\} \cup H, j \in O\}. \end{aligned}$$

For each connection $(i,j) \in C$, let $w(i,j) \in \mathbb{R}$ be a real value, assigning a *weight* to the directed connection between neurons i and j.[4]

Let $a_t(i)$ be the activation of a neuron i at time t. At time 0 the activations are given by

[4]Here, the variables i and j refer to neurons, they are not indexes.

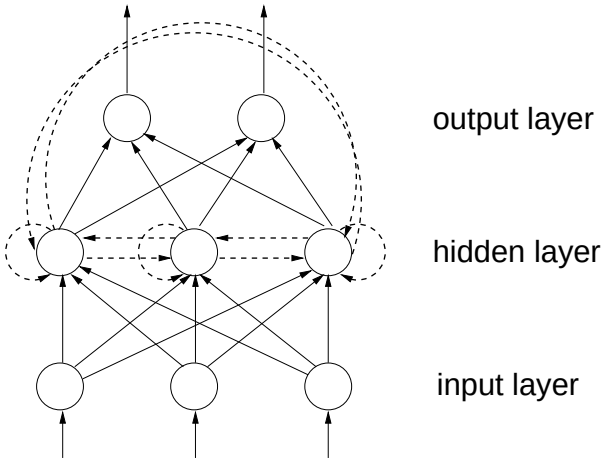

Figure 2.2.: Elman network having an input layer of 3 nodes, a hidden layer of 3 nodes and an output layer of 2 nodes. There are two connection types, *feed-forward* connections (solid lines) and *recurrent* connections (dashed lines). While feed-forward connections pass a change in the activation of the input node to the output node without delay, a recurrent connection always passes the activation of the last iteration. Note that the hidden nodes are fully interconnected by recurrent connections.

$$a_0(j) = \begin{cases} 0 & \text{if } j \in N \setminus \{b\}; \\ 1 & \text{otherwise.} \end{cases}$$

The *bias neuron* b has a constant activation $a_t(b) = 1$, for $t \geq 0$. The activations $a_t(i), i \in I, t > 0$ correspond to the s-bot's sensor reading values at time t.

The activation $a_t(i), i \in H, t > 0$ is given by the following equation:

$$a_t(i) = f\left(w(b,i) + \sum_{k \in I} a_t(k) * w(k,i) + \sum_{k \in H} a_{t-1}(k) * w(k,i)\right),$$

where $f(x)$ is a logistic function:

$$f(x) = \frac{1}{1 + \exp(-x)}.$$

The activation $a_t(i), i \in O, t > 0$ is given by the following equation:

$$a_t(i) = f\left(w(b,i) + \sum_{k \in H} a_t(k) * w(k,i)\right).$$

In the experiments described in this chapter, the neural network controllers have five input neurons, namely prey_x, prey_y, light direction_x, light direction_y and gripper status. In the following the activation of each input neuron at time t is described:

- Let $R = 50$ be the sensing range, d_t the horizontal distance between the s-bot and the border of the prey object at time t, and α_t the horizontal angle of the perceived prey item with respect to the s-bot's heading. In case no prey item is perceived, the provided distance is R. The activation values are given by

$$a_t(\text{prey}_x) = \begin{cases} (R - d_t)\sin\alpha_t & \text{if } d_t < R; \\ 0 & \text{otherwise,} \end{cases}$$

and

$$a_t(\text{prey}_y) = \begin{cases} (R - d_t)\cos\alpha_t & \text{if } d_t < R; \\ 0 & \text{otherwise.} \end{cases}$$

- $a_t(\text{light direction}_x) = a_t(\text{light direction}_y) = 0$ in case no light is perceived. Otherwise, the camera sensor provides the horizontal angle β_t in which the light emitted by the beacon is perceived at time t. The activation values are given by

$$a_t(\text{light direction}_x) = \sin\beta_t, \text{ and}$$
$$a_t(\text{light direction}_y) = \cos\beta_t.$$

- $a_t(\text{gripper status})$ is set to 1 in case the s-bot is connected to another object via the gripper element at time t, otherwise it is set to 0.

The activations of the output neurons are within the interval (0,1). They are used to control the actuators:

- For the desired angular speed of each motorized wheel, the corresponding output is scaled in the range $-15\,s^{-1}$ to $15\,s^{-1}$.
- The gripper actuator tries to grip if and only if the corresponding output is greater than 0.5.

2.3.2. The Evolutionary Algorithm

The evolutionary algorithm used is a self-adaptive version of the $(\mu + \lambda)$-ES (see Section 1.1.3). An individual is composed of real-valued object parameters $X = (x_1, x_2, \ldots, x_N)$ specifying the weights of the Elman Network used to control the s-bots, and real-valued strategy parameters $S = (s_1, s_2, \ldots, s_N)$ specifying the mutation strength used for each coordinate of the search space.

Initially, a population of randomly initialized individuals is generated. Object parameters are set to random values within $[-5, 5]$, and strategy parameters are set to random values within [0.01,1]. In principle, object parameters are not limited to

this interval. Also strategy parameters can overcome the upper bound of 1. However, there is a strict lower bound of 0.01 in order to avoid stagnation: in case of mutating object parameters, the standard deviation should not converge to 0 because mutation would not provide new solutions any more.

Before the evolution is started, a *random walk* strategy is performed in order to generate a wide variety of somewhat acceptable starting solutions. The total number of fitness evaluations during the random walk is equal to number of evaluations of an evolution lasting 10 generations.

In each generation all individuals are assigned a fitness. The μ parent individuals from the previous generation are re-evaluated due to the following reasons:

- The fitness of an individual is affected by noise and by some initial conditions such as the sbots' placements in the environment. It has been observed that strategies of pure average performance may reach very high fitness values in a single try. In case elitism is used in an evolutionary algorithm with a noisy fitness function, the best rated individuals tend to be overestimated. This overestimation can be typical, in case large populations are used or if a considerably big fraction of individuals can get comparatively high fitness values. Re-evaluating parent individuals avoids a systematic over-estimation caused by previous fitness evaluations in time. It is noted that over-estimation can have the positive effect of protecting good solutions (Markon et al. 2001).

- Individuals of the same generation are evaluated under similar conditions, for instance, the initial placement of the s-bots in the environment is identical. For each generation, the conditions are changed randomly. Parent individuals get re-evaluated under the same conditions that are used to evaluate offspring individuals. Therefore, we expect to gain better comparability of the parent and offspring fitness values.

The best μ individuals create λ offsprings. With a probability of 0.8 an offspring is created by mutation, otherwise two parent individuals are selected and recombined. Finally, the offspring is mutated.

Mutation of strategy parameters is performed by multiplying the strategy parameter s_i with a random variable using the lognormal distributional approach with the parameters τ_{g} and τ_{s} (for details see Section 1.1.3). τ_{g} is set to $\frac{1}{\sqrt{2N}}$ and τ_{s} is set to $\frac{1}{\sqrt{2\sqrt{N}}}$. Then, the object parameters x_i are mutated by adding a random variable according to the normal distribution $N(0, s_i)$.

Recombination does combine two parents individuals by intermediate or discrete recombination (see Section 1.1.3), both with same probability.

The number of offspring has been chosen as $\lambda = 80$ and the number of parents is $\mu = 20$. Since the amount of parents in the total population is only 20%, there is

substantial *selection pressure*. On the other hand, only 20% of fitness evaluations is spent to re-evaluate parent individuals in the new generation.

2.3.3. The Fitness

The task is to control a small group of simple s-bots to cooperate to transport as far as possible a heavy prey within a fixed time period. The direction of movement of the prey is not predetermined.

The prey is modeled as a cylinder of radius 12 cm, height 10 cm and 500 g mass. The prey cannot be moved by a single s-bot. Therefore, co-ordination is necessary although the s-bots cannot sense each other.

Each individual, i.e., a common controller for a group of s-bots, is evaluated by performing tests $t_1, t_2 \ldots, t_S$ against a sample composed of S configurations specifying the s-bots' initial placements and the position of the beacon. $S = 5$ can be called the sampling size. The sample is changed only once a generation, therefore, all individuals that compete with each other are evaluated under similar conditions[5]. This should increase the comparability of fitness values among individuals within the same generation. Note that the μ parent individuals that are copied into the next generation by default get re-evaluated based on the new sample.

In each test, the simulation lasts 20 seconds. The prey item is placed in the center of the environment. The s-bots are placed at random positions and orientations, but not more than 25 cm away from the prey. This ensures that the prey can initially be detected by each s-bot. The beacon is placed at a random position 300 cm away from the center of the environment. This is is less than the distance the prey can be moved within the simulation time of 20 seconds.

For each test $t_i, i \in \{1, \ldots, S\}$ a quality measure q_i evaluates the exhibited transport performance. The final fitness score f is computed using a weighted average. Let ϕ be a permutation of $\{1, 2, \ldots, S\}$ such that $q_{\phi(1)} < q_{\phi(2)} < \cdots < q_{\phi(S)}$, then the fitness value is given by

$$f = \sum_{i \in \{1,\ldots,S\}} (S - i + 1)\, q_{\phi(i)}. \tag{2.1}$$

In this way, tests exhibiting weak transport performance account more than others. Due to the weighting, any fluctuation gets punished. Therefore, individuals that emerge during evolution should get more and more resistant to the noise in the environment, their sensors and actuators. This strategy emphasizes solutions with low performance fluctuations. Having a high average performance is not the only aim. It is desired to have a group of s-bots that is almost always able to complete the transport task with an acceptable performance.

[5] Settings concerning noise acting on sensors and actuators are not fixed by the sample's configurations.

To emphasize the evolution of solutions that let all s-bots participate in the transport activities, the quality measure accounts for the clustering performance of s-bot's around the prey. The clustering performance c_i of test i is defined by

$$c_i = \sum_{j=1}^{n} c_i^{(j)}, \text{ and } c_i^{(j)} = \begin{cases} 0 & \text{if } d_i^{(j)} > 50; \\ 1 & \text{if } d_i^{(j)} < 25; \\ \frac{50 - d_i^{(j)}}{25} & \text{otherwise.} \end{cases} \tag{2.2}$$

For test t_i, $d_i^{(j)}$ gives the Euclidean distance between the final positions of the jth s-bot and the center of the prey minus the radius of the prey.

Therefore, s-bots that are not more than 25 cm away from the perimeter of the prey get the maximum clustering performance value of 1. In this way, any structure of two collaborating s-bots pushing or pulling the prey item cannot be punished. S-bots that are more than 50 cm away cannot sense the prey item any more and get the lowest possible clustering performance value of 0.

The quality measure q_i is defined as

$$q_i = \begin{cases} c_i & \text{if } D_i = 0; \\ 1 + (R + \sqrt{D_i}){c_i}^5 & \text{otherwise,} \end{cases} \tag{2.3}$$

where D_i is the Euclidean distance between the prey's initial and final positions, and $R = 1$ is a constant reward.

In case the prey item has not been moved in test t_i, the quality measure q_i is given by the clustering measure $c_i \in [0, 1]$ only. Otherwise, the fitness is at least 1. In this case the clustering measure again has an important influence. Therefore, it is very likely that successfully evolved controllers encourage the s-bots to remain in the vicinity of the prey.

The term $R + \sqrt{D_i}$ is composed of $R = 1$, a small, distance independent reward, and the root of the distance D_i. D_i should not exceed a distance of 150 cm (see next section). The root function is applied as the scaling function in order to emphasize small differences for lower distance values and to weaken differences for higher distance values.

2.4. Results

The experimental setup described above has been used in 10 independent evolutionary runs. Each one was performed for 150 generations. One run lasts a little bit less than a week on a machine equipped with 512 MB of memory and a 1,537 MHz processor[6]. In the following, the quality of transport as well as the behaviors and their ability to scale using larger group sizes are discussed.

[6] AMD Athlon XP™ 1800+

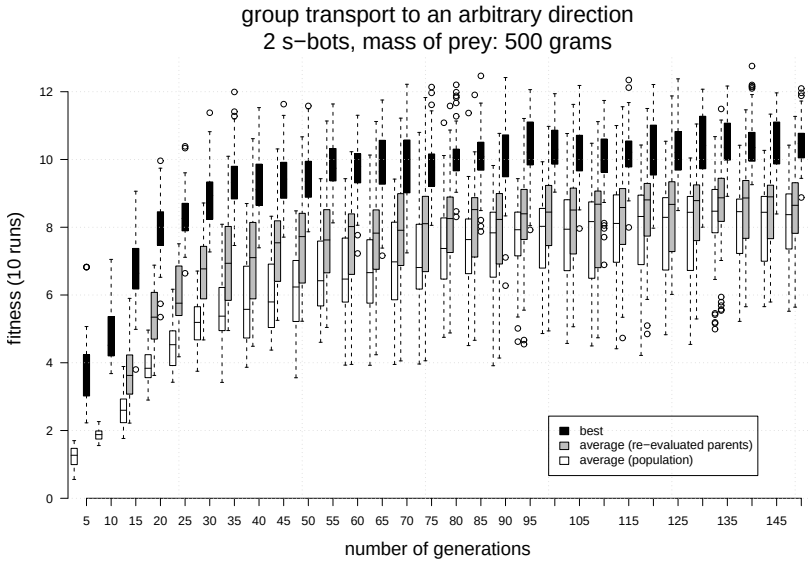

Figure 2.3.: Box-and-whisker plot (see also description of Figure 2.5) providing information about the evolutionary progress in 10 runs. Characteristics about three types of observations are displayed (partitioned into groups of 5 generations): the best fitness value of a population (black boxes), the average fitness value (white boxes), and the average fitness value of the re-evaluated parent individuals (grey-boxes). Since a random walk instead of the evolutionary algorithm has been applied just for the first 10 generations, two boxes concerning the average of the parent individual's fitness are missing.

2.4.1. Performance and Robustness

The quality measure of the group's transport performance in a test depends on the genotype specifying the neural network controller, the s-bots' initial positions and orientations, the position of the beacon in the environment, the particular offset and threshold values of each sensor and actuator, and the amount of noise that affects sensors and actuators at each time step. Of course, the ultimate goal is to generate genotypes that perform well under every potential condition. However, the number of different conditions is uncountable, and during evolution the solutions are evaluated using just a sample of 5 different conditions per generation.

Figure 2.3 displays the development of the best and average fitness for certain sets of individuals over all 10 evolutionary runs:

- The black boxes correspond to the observed fitness values of the best rated individuals for every population. Each box covers a period of 5 generations. As discussed in Section 2.3.2, these fitness values tend to over-estimate the corresponding individuals.

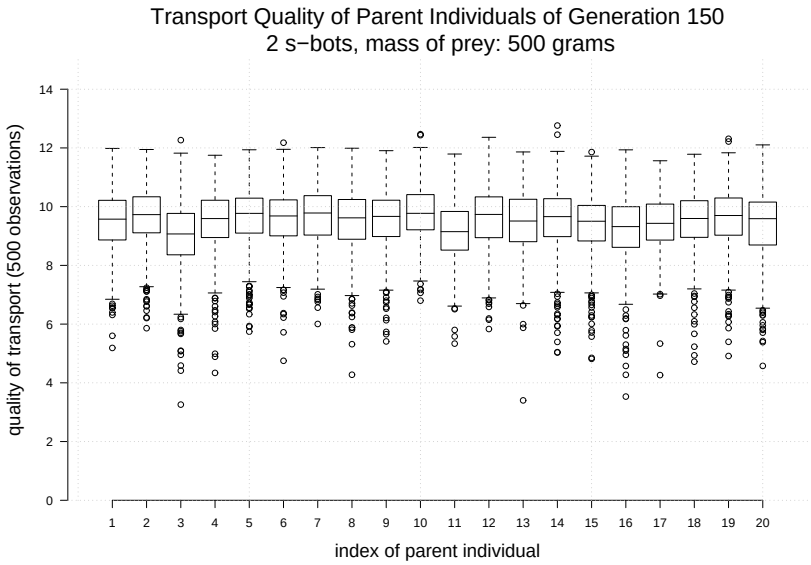

Figure 2.4.: Box-and-whisker plot (see also description of Figure 2.5) visualizing certain characteristics of the transport quality of the μ best individuals of the final generation of one particular run. All parents exhibits similar performance.

- The white boxes correspond to the average fitness values of the entire population.
- The average fitness values of the μ parent individuals of each population are illustrated by the gray boxes. Parent individuals get re-evaluated in the subsequent generation (for reasons see Section 2.3.2). Here, the average fitness value of the parent individuals is computed based on the set of re-evaluated fitness values. Although the average fitness of the re-evaluated parent individuals may fluctuate, there is no systematic over-estimation caused by previous fitness estimations, since the re-evaluation takes place after selection is applied.

Looking at Figure 2.3 one can see that, in all cases, the fitness values tend to increase. Within the first 60 generations, often, the parent individuals exhibit a much higher fitness in average than the entire population of individuals.

Since the fitness values are computed using a weighted average, it is hard to estimate the attained quality level or standard deviation. Moreover, the fitness values are affected by noise. It's not obvious what quality level has been reached and which individuals can be considered *best*.

The set of the $\mu = 20$ parent individuals of a final generation comprises all genetic material that would be exploited in subsequent generations in case the evolution would be continued. Therefore, the μ best individuals of the final generation of all 10

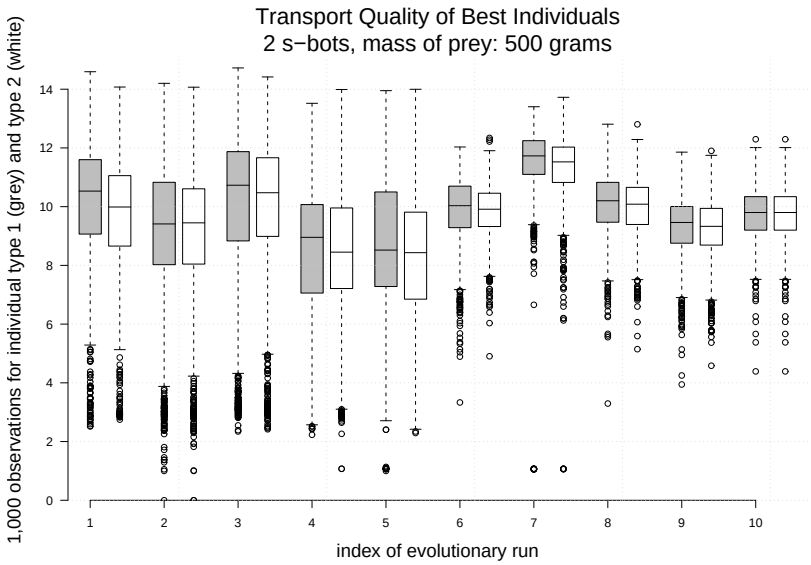

Figure 2.5.: Box-and-whisker plot visualizing certain characteristics of the transport quality of the best individuals of each final generation. The box comprises observations ranging from the first and third quartile. The median is indicated by a special line, dividing the box into the upper and lower part. The whiskers extend to the most extreme data point which is no more than 1.5 times the length of the box away from the box. Data points which lie beyond the extremes of the whiskers are marked as circles.

evolutionary runs are post-evaluated on a sample of 500 different tests. Performing this post-evaluation, it has been observed that parent individuals of the final generation of the same evolutionary run exhibit a quite similar performance; one example is provided by Figure 2.4. Therefore, focus is given on two types of individuals from each evolutionary run that are considered *best*: the one with the highest average performance (type 1), and the one with the lowest observed standard deviation (type 2). Both types of individuals are post-evaluated for a second time, on a random, but fixed set of 1,000 tests. In Figure 2.4 the individual with the highest average is equal to the one with the lowest observed standard deviation. It is the individual with index 10.

Figure 2.5 shows a box-and-whisker plot presenting the observed quality of transport in the 1,000 tests for both types of best individuals for all evolutionary runs. The first five solutions displayed make an essential use of the gripper element.

According to Equation 2.3, the quality measure q_i of try i is $1 + (R + \sqrt{D_i})c_i{}^5$ and therefore at least 1 in case the prey object has been moved, and $c_i \in [0, 1]$ otherwise. Looking at Figure 2.5, it can be seen that the prey has been moved in almost all cases, since almost all observed quality measures are in the range 2 to 15. In case

$q_i \geq 2$, the distance D_i the prey has been moved is at least $(q_i - 2)^2$. In case of $c_i < 1$, D_i is bigger than $(q_i - 2)^2$. Assuming the worst case for the distance moved, that is, a perfect clustering value of $c_i = 1$, some examples for pairs of moved distance D_i and the corresponding quality of transport q_i are given in the following table:

q_i	1	3	4	5	6	7	8	9	10	11	12	13	14
D_i	0	1	4	9	16	25	36	49	64	81	100	121	144

In case two s-bots are placed next to the prey, forming a pulling chain and applying the maximum speed to the wheels, the distance the object can be moved within the simulation time of 20 seconds is around 149.64 cm. This corresponds to a transport quality of approx. 14.23. A pushing chain of two s-bots exhibits an equal transport quality. During these measurements, no noise function has been applied to the wheels. If noise is not discarded, the wheel's speed is fluctuating around the desired setting, leading to similar performances for a pulling chain, and lower performances for a pushing chain.

However, during the fitness evaluation the s-bots are not placed in such an initial structure that can be used to retrieve the prey: the s-bots are scattered in the environment at random positions and with random orientation. Therefore, the s-bots have to approach the item and to co-ordinate themselves in some way, before a structure is formed that can be used to retrieve the item. If we assume that the randomly placed s-bots need 10 seconds to form a chain like the one described, this structure could only pull the prey for the remaining 10 seconds of simulation time. During this period, the item can be moved 74.50 cm, corresponding to a transport quality of approx. 10.63. In case only 5 seconds are available to move the prey, a distance of 36.90 cm can be reached, corresponding to a transport quality of approx. 8.07.

The median performance exhibited by half of the type 1 individuals is in range [10.03,11.73]. The observed standard deviation concerning half of the type 2 individuals is in range [0.91,1.30].

Overall, we have evolved a number of controllers of respectable performance. Some of them exhibit only low performance fluctuations. Therefore, the controlled groups of s-bots act quite robustly with respect to various kinds of noise concerning the sensors and actuators devices.

In the following, the obtained behaviors are briefly described.

2.4.2. Behavioral Analyses

The evolutionary runs have generated a variety of high performing solutions. In case of the 5 solutions on the left side of Figure 2.5 the gripper element is an essential part

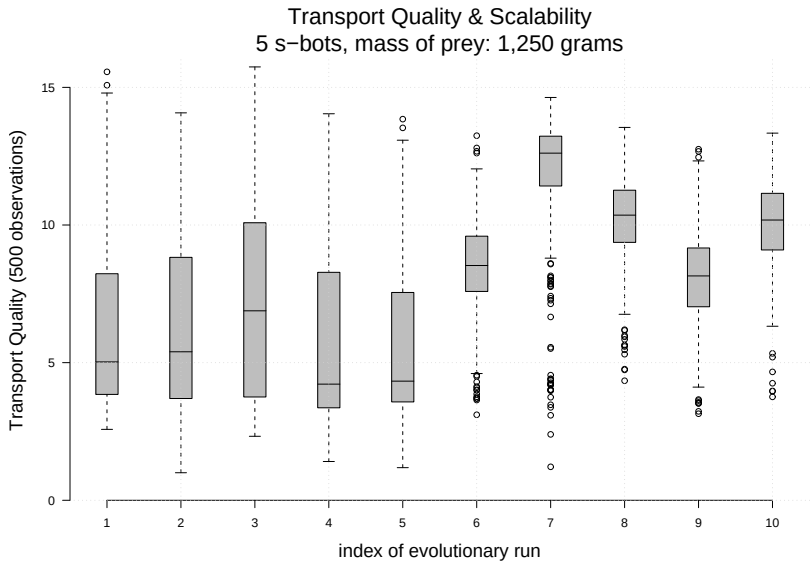

Figure 2.6.: Scalability test: for each evolutionary run, the individual for which the best average performance has been observed (the type 1 individual) is evaluated 500 times using a group of 5 s-bots and a prey of radius 30 cm and mass 1,250 g. The simulation period is extended by 10 more seconds, since it requires more time for each s-bots to orbit around the prey. The first five individuals make an essential use of the gripper element.

of the transport strategy. The performances of these solutions are more fluctuating than the performances of solutions not relying on the gripper element.

Some solutions let the s-bots drive forward in order to push the prey. In this case, the gripper element is used to establish a connection and sometimes even a pushing chain of 2 s-bots moving like a snake is formed. Other solutions let the s-bots drive backwards. In this case, the s-bot's torso is in direct contact with the prey, since the gripper element is heading in the other direction. In one case, solutions even combine both strategies: s-bots may push the prey driving in both orientations. None of the best solutions exploits gripping in order to pull the prey.

Usually, s-bots orbit around the prey until they perceive a certain direction of light. Then they start trying to move the prey. In most solutions the prey is moved towards the light, or in the opposite direction. In one evolutionary run, solutions let move the prey perpendicular to the direction of light perceived, therefore, the prey circuits around the beacon. Within the short simulation period, this behavior is not punished.

The question arises whether the behaviors are scalable, i.e., are evolved individuals able to cooperate in transport of a heavier prey using a larger group size?

2.4.3. Scalability

For each evolutionary run, the individual with best average performance is evaluated using a prey of mass 1,250 g and a group of 5 s-bots. In most cases, the exhibited performance is weak. Having 5 s-bots around a prey of same size seems to increase the inter-individual competition for the limited space in which the desired direction of light is perceived.

However, in case the perimeter of the prey is multiplied by the same factor the number of s-bots is increased, all solutions are able to move the prey, and the ones not relying on the gripper exhibit an acceptable performance (see Figure 2.6).

2.5. Conclusions

In this chapter we addressed the problem to control a group composed of 2 simple s-bots to cooperate to transport as far as possible a heavy prey of specific shape, size and mass within a fixed time period into an arbitrary direction. The prey cannot be moved without cooperative behavior. There is no explicit communication among s-bots. Moreover, the s-bots are not able to sense each other. The communication structure is limited to *interactions via the environment*.

In all of the 10 evolutions that have been carried out, controllers exhibiting an acceptable performance under most of the tested 1,000 conditions have been generated. In half of the runs controllers reach high performance levels while having quite low performance fluctuations. Moreover, many solutions can be applied to larger groups to move prey items of bigger size and mass. In general, the controllers' performances are very sensitive with respect to the size of prey. Further on, we observed that the performances of those solutions that make essential use of the gripper element are more fluctuating than the performances of other ones.

Some solutions perform quite robustly with respect to various kinds of noise concerning the sensors and actuators devices. However, since only two s-bots are available, the system cannot complete the task if one of them fails.

Chapter 3.

Evolving Cooperative Transport Strategies for small Swarms

In the previous chapter we have seen that we can evolve neural networks able to control a small group of two s-bots in order to move a prey of specific shape, mass and size as far as possible into an arbitrary direction. The evolved controllers act very efficient on the vast majority of 1,000 different conditions concerning the s-bots' initial placement and behavioral deviations of sensor and actuator devices. The best evolved solutions have been shown to perform fairly well when applied to larger groups of s-bots moving bigger and heavier prey.

However, we have seen that the evolved solutions exhibit poor performances in case there is too much inter-individual competition for the limited space around the perimeter of prey. It seems that the strategies are not able to let the s-bots form functional structures of several layers in order to pull or push the prey. Therefore, small but heavy prey object are not moved successfully.

In the experiments presented in this chapter the utilization of the gripper element and the construction of structures is emphasized:

- During fitness estimation, the controller to be evaluated gets a reward M for each s-bots being part of a self-assembled structure in case the structure is connected to the prey. However, the same reward is realized, in case a solitary s-bot connects to the prey. There is no reward for self-assembled structures that are not connected to the prey. Like that, we expect that direct connections to the prey are likely to be established. Structures of self-assembled s-bots may occur in case of the particular prey item is small enough.

- The s-bots are provided with a new rotational degree of freedom to rotate the turret (including the gripper element) with respect to the chassis. Therefore, s-bots that are connected to other s-bots or to the prey retain more flexibility.

- Finally, the s-bots that are used in the experiments described in this chapter can sense the proximity of their nest mates. This may give an advantage when building structures composed of multiple s-bots.

In this chapter we apply an artificial evolution to synthesize neural networks in order to control a group composed of 4 simple s-bots engaged in a cooperative transport task. Again, we consider the case that only one single prey is present. Furthermore, no advanced search or recruitment techniques are necessary, since the s-bots are placed in the immediate vicinity of the prey. We assume that all s-bots can be guided from a potential nest to the prey, e.g., by following a path established by a group of specialized s-bots connecting the nest with the prey.

Two sets of experiments are performed:

- in the first set, the prey object is a cylinder of a specific shape (radius 8 cm and height 10 cm). Since the prey item is rather small, we expect that some of the evolved controllers will let the group of s-bots form structures of assembled s-bots to move the prey.

- in the second set, the prey object is modeled by cylinders and cuboids of various shape, size and height. Depending on the height, prey objects may shadow the target direction indicated by a constant light signal emitted by the beacon. We expect that this task is more difficult, due to high varying sizes and shapes of the prey.

As in the previous experiments, there is no explicit communication between s-bots. However, *interactions via sensing* are possible (see Section 1.3.3). The s-bots can sense a beacon that is placed in the environment as long it is not shadowed by any other object. To be able to recognize the surface of prey objects of various shape and size, a new type of prey sensor (realized by the camera device) is introduced. This sensor can be actively controlled by the s-bot's control system in order to scan the surrounding for prey. A similar approach is utilized in order to perceive other s-bots in the vicinity.

The following two sections introduce the underlying simulation framework and the task that is studied. In Section 3.3 the experimental setup is detailed. Section 3.4.1 is devoted to survey the skills and potential weaknesses of the new prey sensors introduced. In Section 3.4 the results of the main experiments are presented.

3.1. Simulation Framework

Certain aspects of the simulation model have been modified for two reasons:

- to cope new information available about the developing prototype of the real s-bot, and

- to exploit the manifold abilities of the real s-bot's sensors and actuators.

Table 3.1.: Friction coefficients belonging to pairs of material types in contact. The "-"-symbol indicates that no contact of the corresponding pair of material types will occur in the experiments described in this chapter.

	ground	wheels	torso	prey
ground	-	0.54	-	0.25
wheels	0.54	0.2	0.2	0.2
torso	-	0.2	0.2	0.2
prey	0.25	0.2	0.2	-

In addition prey objects of various shape, size and mass are available in simulation.

The friction coefficients concerning possible contacts among material types are presented in Table 3.1. Since in the new model, the s-bot's passive wheels are not fully covered by the chassis (see Figure 3.2 and Figure 1.2), the wheels can get into contact with objects other than the ground.

3.1.1. The Prey

Prey objects are modeled as cylinders or cuboids of different size and mass. There are prey objects of two different heights: prey taller than an s-bot (20 cm height) and prey smaller than an s-bot (12 cm height). Tall prey objects are of special interest since they may prevent any visual contact between two s-bots or between an s-bot and a beacon. Objects of height 12 cm have a uniform mass distribution. Taller objects are modeled as a stack of two objects of different mass (see Figure 3.1). The mass of the object on the top of the stack is set to 25% of the total mass. This permits the use of tall prey objects with relative small bases that do not topple down when pushed or pulled by the s-bots.

In order to model an accurate rotational behavior of a prey object of a certain distribution of mass, it is important to properly model its *moment of inertia* in simulation. This is detailed in Appendix A.

3.1.2. The S-Bot

The new s-bot model is shown in Figure 3.2. The main difference to the previous one (see Section 2.1.2) is that the new s-bot model provides another degree of freedom that enables the rotation of the s-bot's upper part (the turret) with respect to the lower part (the chassis) by means of a motorized axis (Mondada et al. 2002).

The chassis provides the support for the two passive and two active wheels. Concerning the wheels' physical properties and alignment, as well as the ground clearance the new model corresponds to the previous one. The chassis is modeled by a cylinder of diameter 4.5 cm, height 3.75 cm and mass 300 g.

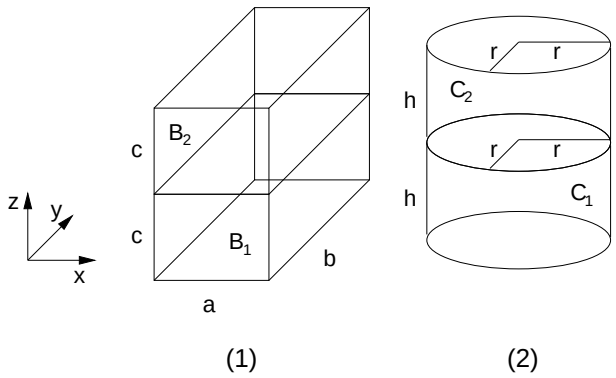

Figure 3.1.: Prey objects composed of a lower part with mass m_1 and an upper part with mass m_2. Each part has a uniform density. (1) cuboid and (2) cylinder.

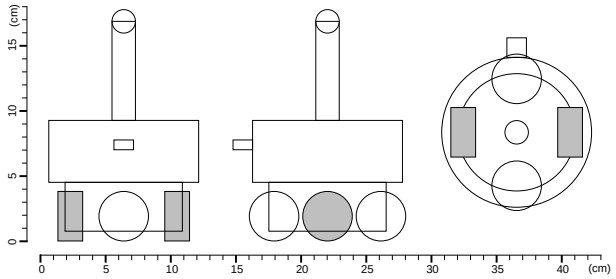

Figure 3.2.: Front, side and top view of the s-bot: the upper part (the turret) can rotate with respect to the lower part (the chassis) by means of a motorized axis. The turret is composed of a cylindrical torso, a small gripper element heading to the front, and a camera placed on a pillar. Two passive, spherical wheels (white) are connected to the torso's back and front part. Two motorized, cylindrical wheels (grey) are connected to the torso on the left and right side. Collisions between wheels and the torso are disabled.

The turret consists of a cylindrical torso, the gripper element that is fixed on the torso's edge heading to the s-bot's front, and a long cylindrical pillar placed on the top of the torso to support a visual camera system. The torso is modeled by a cylinder of diameter 11.5 cm, height 4.75 cm and mass 200 g. The gripper element and the pillar have a mass of 40 g each. The whole s-bot has a height of 16.85 cm and a total mass of 660 g.

In the following two sections the actuators and sensors of the s-bots are detailed.

Actuators

The s-bot's actuators for the active wheels and the gripper element are already described in Section 3.1.2. The only difference to the previous model concerns the gripper element: to model the level of stability achieved with the last prototype of

Table 3.2.: List of information provided by those sensors that are used in the experiments described in this chapter.

Sensor device	Information provided
gripper status sensor	status of being connected to another object via the gripper element
camera sensor	prey proximity, i.e. the distance to the first intersection of a sensing ray oriented to a specified horizontal angle with a visible prey object (sensing range: 50 cm)
	s-bot proximity, i.e. the distance to the first intersection of a sensing ray oriented to a specified horizontal angle with a visible s-bot (sensing range: 50 cm)
	horizontal angle the sensing ray is oriented to
	horizontal angle of highest intensity of light perceived (a constant light signal is emitted by a beacon)
rotation sensors	angular offset between turret and chassis

the gripper element of the real s-bot, the joint that is created by the simulated gripper element can transmit forces up to $30N$. In case higher forces occur the connection will be released automatically (see Section 2.1.2).

The s-bot's turret can rotate with respect to the chassis since both bodies are linked with a motorized hinge joint. The torque that can act in order to rotate the turret to a given angle with respect to the heading of the chassis is limited by 750,000 dyne cm. The maximum angular speed is $2\,s^{-1}$.

To simulate the inaccurate behavior known from real actuators, the desired angle of rotation is slightly modified by adding a random variable which follows the normal distribution $N(0, 0.0025)$.

Sensors

In the experiments described in this chapter, the s-bot is equipped with a *gripper status sensor*, a *rotation sensor* to perceive the orientation of the turret with respect to the chassis, and an omnidirectional camera. The information provided by these sensors is summarized in Table 3.2.

The rotation sensor provides the current angular offset α_{rot} between the orientations of the turret and the chassis. Since it might take up to several seconds to reach a desired orientation of the turret, the current offset value that is perceived by

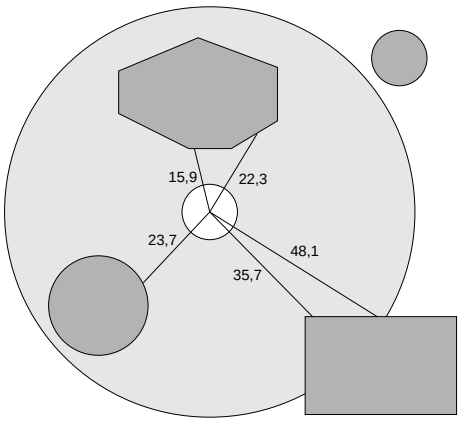

Figure 3.3.: Mode of action of *prey proximity sensor*: top view on a 2D environment with four different prey objects (grey) and an s-bot (white circle) placed in the center. The s-bot's sensing range of 50 cm is indicated by the light gray circle. The camera sensor provides a default value in case no prey is perceived in the direction specified. Otherwise, it provides the horizontal distance to the closest position in which the object and the sensing ray intersect. Five example rays all having an intersection are displayed. The rays are labeled with the corresponding distances (without being affected by noise).

this sensor can differ from the value the rotational actuator is set to. α_{rot} (radian measure) is modified slightly by adding a random variable which follows the normal distribution $N(0, 0.0025)$.

The camera sensor has already been introduced in the description of the previous experiments (see Section 2.2). In the s-bot model that is described in this chapter, the camera is fixed on the turret that can rotate with respect to the chassis. We assume that we can extract features like angular positions of perceived objects with respect to the heading of the s-bot's chassis, since the rotational offset between the chassis and the turret is available.

In the set of experiments presented in the previous chapter, the camera sensor has provided

- the distance to the closest point that belongs to the edge of a prey object, and
- the angle concerning the direction to that particular point with respect to the chassis' orientation.

Since only cylindrical prey objects have been utilized, the direction to the closest point of the prey object's edge, is also given by the direction to the center of the prey.

Since we are interested in transporting prey objects of various shapes, we equip the simulated s-bot with a new prey sensor that is based on the camera sensor again.

The idea is to provide the s-bot with the ability to scan the edge of a prey. The s-bot should be able to scan for prey into different directions. This ability should be available also in case the s-bot has lost its mobility, since it is part of an assembled structure.

The camera sensor can provide the horizontal distance to the next visible prey concerning a specified angle in the horizontal[1] plane (see Figure 3.3) - we also refer to this using the term *prey proximity sensor*. The five rays illustrated in Figure 3.3 correspond to five sensor readings - each time the prey proximity sensor is aligned into a different horizontal direction. The sensing range is limited to objects not more than 50 cm away from the s-bot.

Active perception that is the process of selecting sensory patterns by motor actions can turn *hard* discrimination problems into *simpler* ones (Nolfi 1998). Further on, Nolfi has studied a system using a 3-segments arm with 6 degrees of freedom in order to discriminate a spherical and a cubic object (Nolfi and Marocco 2002). Within the context of vision processing, "feature-selection mechanisms and behavior of autonomous robots" have been co-evolved for shape discrimination and navigation tasks (Kato and Floreano 2001; Marocco and Floreano 2002).

In the experiments described in the this chapter, s-bots are able to sense each other. Analogical to the prey proximity sensor, the camera sensor can provide the horizontal distance to the next visible s-bot object concerning a specified angle in the horizontal plane. This is referred also by the term *s-bot proximity sensor*.

The sensor readings of the s-bot/prey proximity sensor are affected by various types of noise. The specified angle that indicates the sensing direction is modified slightly by adding a random variable of the normal distribution $N(0, 0.0025)$. In case an object is perceived, the distance is multiplied by a random variable which follows the normal distribution $N(1, 0.02)$. Further on, the sensing range of each s-bot is modified once for the whole simulation period by adding a random variable of the normal distribution $N(0, 1)$.

3.2. The Task

The Objective Our aim is to control a group composed of 4 simple s-bots to cooperate to transport a heavy prey within a fixed time period as close as possible to a light emitting beacon. Depending on the mass that is varying among different prey objects, the prey can be moved efficiently only by cooperative behaviors of two, three or all four s-bots.

The Environment The s-bots' environment consists of a flat ground. The prey is placed in the center. A light emitting beacon is placed 300 cm away. The s-bots are

[1] With respect to the body frame of the s-bot's chassis that is the s-bots own coordinate system.

placed at random positions and orientations within a semi-circle around the prey (see Section 3.3.3).

The S-bot's Capabilities The s-bot can actively scan its surrounding for prey and teammates using the prey and s-bot proximity sensors (see Section 3.1.2) that are aligned in a common direction specified by the s-bot's control system (see Section 3.3.1). Moreover, an s-bot can sense the beacon as long it is not shadowed by the prey or other s-bots. The s-bot is able to control its gripper elements and it perceives whether a connection is established or not. Further, it is able to rotate its turret with respect to the chassis. The s-bot can sense the current angular offset between the turret and chassis.

The sensors and actuators are affected by various types of noise. Moreover, sensor and actuator devices of same type behave not identical.

3.3. Experimental Setup

3.3.1. The Neural Network Controller

The architecture used in order to control the s-bots is a recurrent neural network evolved by an evolutionary algorithm. Each s-bot of a group acting in a simulation environment is initially equipped with an identical neural network.

To control the s-bots, an Elman Network like the one used in the previous set of experiments is utilized (see Section 2.3.1). Elman networks are recurrent neural networks having a fully interconnected hidden layer of neurons.

The network used in the set of experiments described in this chapter is illustrated in Figure 3.4. In the following the network inputs and outputs are explained.

The neural network controllers have six input neurons corresponding to the current sensory information, i.e. the gripper status, the current rotation of the turret, the horizontal light direction, the prey and s-bot proximity, and the angle the proximity sensor is set to (see Figure 3.4). In the following the activation $a_t(\cdot)$ of each input neuron at time t is described:

- $a_t(\text{gripper status})$ is set to 1 in case the s-bot is connected to another object via the gripper element at time t, otherwise it is set to 0.
- Reading values from the rotation sensor are in range $[-\pi,\pi]$. To compute the activation $a_t(\text{turret rotation})$, the reading value is linearly scaled to [0,1].
- $a_t(\text{light direction})$ is set to 0 in case the light emitted by the beacon is not perceived at time t. This is possible because there might be an object placed between the beacon and the camera. Otherwise, the camera sensor provides the horizontal

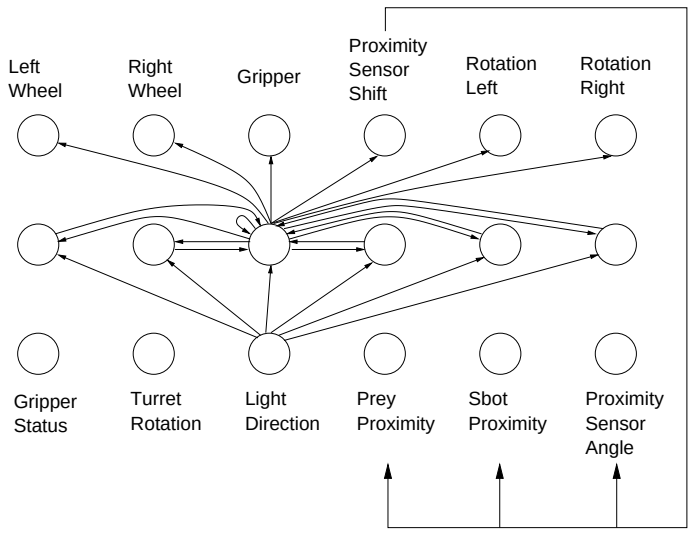

Figure 3.4.: Elman network having an input layer of 6 nodes, a fully interconnected hidden layer of 6 nodes and an output layer of 6 nodes. All connections from and to the third node of each layer are visualized. The output labeled *Proximity Sensor Shift* is used to move the proximity sensor to a new horizontal angle. The inputs *Prey Proximity* and *S-bot Proximity* are the readings of the proximity sensor. The current angular setting of these sensors is also provided by the input *Proximity Sensor Angle*.

angle $\beta_t \in [0, 2\pi]$ in which the light emitted by the beacon is perceived at time t. The activation value is given by

$$a_t(\text{light direction}) = 0.5 + 0.5\frac{\beta_t}{2\pi}.$$

- If the prey proximity sensor does not perceive any prey in the specified direction at time t, the corresponding activation value is set to 0. Otherwise, $a_t(\text{prey proximity}) \in [0.1, 1.0]$: let d_t be distance provided by the prey proximity sensor at time t. Then the activation value is given by

$$a_t(\text{prey proximity}) = 1.0 - 0.9\frac{d_t - r}{R - r},$$

where R is the sensing range that is around[2] 50 cm, and $r = 5.75$ cm is the minimum distance given by the radius of the cylindrical turret.[3]

[2]The sensing range of each s-bot is slightly different. Moreover, it is affected by noise (see Section 3.1.2).

[3]Since the experiments are carried out on a flat plane, and since the gripper element is not able to elevate, it's not very likely that s-bot's bend that much. Therefore, we can assume that the

- The calculation of the activation of the input neuron corresponding to the s-bot proximity sensor is analog to the one of the prey proximity sensor. Both sensing devices are aligned in a common horizontal direction that is determined by the sequence of outputs of the neural network (see below and see Figure 3.4).
- The (common) angle the prey and s-bot proximity sensors are oriented to at time t, serves as an input to the neural network, too. To compute the activation value of the corresponding input neuron, the sensor reading is linearly scaled from the range $[-\pi,\pi]$ to [0,1].

The activations of the six output neurons $o_1, o_2, \ldots, o_6$ are used to control the actuators:

- At time t, the left wheel is set to a desired speed of $(16\, a_t(o_1) - 8)\, s^{-1}$ and the right wheel is set to a desired speed of $(16\, a_t(o_2) - 8)\, s^{-1}$.
- The gripper actuator tries to grip at time t if the activation $a_t(o_3)$ exceeds 0.5. Otherwise, the gripper element does not try to grip.
- The desired rotation γ_t of the turret with respect to the chassis at time t is given by

$$\gamma_t = (a_t(o_4) - a_t(o_5))\pi.$$

- Let δ_t be the horizontal angle of the prey/s-bot proximity sensor at iteration t; by default δ_0 is 0. The new angle δ_{t+1} of the prey/s-bot proximity sensor is given by

$$\delta_{t+1} = \delta_t + a_t(o_6)\frac{\pi}{5} - \frac{\pi}{10}.$$

  Therefore, the proximity sensors can be shifted not more than 18 degree within one iteration (every 100 ms).

3.3.2. The Evolutionary Algorithm

The evolutionary algorithm used is almost identical to the one described in Section 2.3.2. There are only two differences: the first one is that the random walk for the initial 10 generations is skipped. The second difference is that λ is chosen to be 60. Therefore the populations size $\mu + \lambda$ is reduced from 100 to 80.

distance d between the s-bot's center and another object is usually not smaller than the radius r of the s-bot's turret. In case the distance d is less than r, d is set to the lower bound of r.

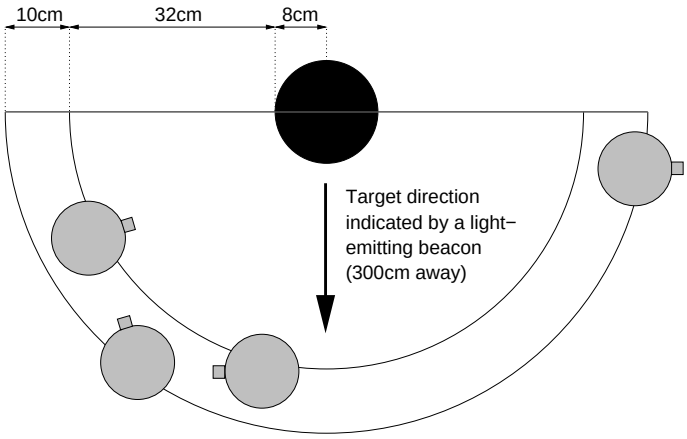

Figure 3.5.: Example of initial placement: the prey object (black) is a cylinder with radius 8 cm. It is placed in the center and it has to be moved in the direction of a light emitting beacon (downward, as indicated by the arrow). Four s-bots (gray) are placed at random positions within a semi-circle, at least 40 cm and at most 50 cm away from the center.

3.3.3. The Fitness

The task is to control a group composed of four simple s-bots to cooperate to transport as far as possible a heavy prey within a fixed time period. The desired direction of movement of the prey is indicated by a beacon that is placed in the environment. The prey object is a cylinder of height 10 cm and radius 8 cm. The mass of prey is varying in the range 500 to 1,000 g. Depending on the mass, the prey can only be moved efficiently if two, three, or if all four s-bots co-ordinate their activities.

Each individual, i.e., a common controller for a group of s-bots, is evaluated by performing five tests $t_1, t_2 \ldots, t_5$ against a sample composed of 5 configurations. Each configuration specifies the s-bots' initial placements, the mass of the prey, and that part of the noise (affecting sensor and actuator devices) that is fixed for the whole life-time of 20 seconds of simulation time. For each sample, the preys' masses of the five configurations are $500, 625, 750, 875, 1{,}000$ g. The sample is changed only once a generation. Therefore, all individuals that compete with each other are evaluated under similar[4] conditions.

In each test, the simulation lasts 20 seconds. The prey item is placed in the center of the environment, the beacon is placed at a distance of 300 cm. The s-bots are placed with random orientation at random positions 40 to 50 cm away from the center. In principle, the s-bots are supposed to be guided from the nest to the prey. In that case, they tend to reach the prey from a similar direction the prey should

[4]There is also noise acting on sensors/actuators that changes each time step. This type of noise is not determined by the sample's configurations.

be moved to. Therefore, the s-bots are placed within a semi-circle that is oriented toward the target (see Figure 3.5).

Initially, the prey object is just within the s-bot's sensing range of 50 cm. However, it would take several seconds for the s-bot's controller to scan successively the entire surrounding with a sufficient accuracy for the small and distant prey object, since the controller is activated only 10 times per second. Therefore, the proximity sensor is initially set to an orientation towards the center[5] in order to simplify localization and approach. The horizontal angle the prey sensor is placed to is affected by noise: a random variable that follows the normal distribution $N(0, 0.1)$ is added to the angle (radian measure). During the first second of simulation time, the proximity sensor is oriented towards the prey. For the remaining period of 19 seconds the sensor direction is controlled by the output of the neural network (see Section 3.3.1).

The beacon is placed 300 cm away from the center of the prey. This is less than the distance the prey can be moved within the simulation time of 20 seconds.

For each test $t_i, i \in \{1, \ldots, 5\}$ a quality measure q_i evaluates the exhibited transport performance. The final fitness score f is computed as the average of all quality measures of the sample. Different from the fitness estimation method used in previous experiments (see Section 2.3.3), no weighted average techniques is used to compute the final fitness score: since the masses of the prey objects range from 500 to 1,000 g, the performances values are expected to fluctuate within the sample. The weighted average described in Section 2.3.3 was chosen in order to punish any fluctuation and to emphasize the evolution of robust controllers with respect to the system's noise and the s-bots initial placement.

The main goal to be achieved by the group of s-bots is to move the prey object as much as possible towards the beacon. In this set of experiments we want to emphasize the emergence of self-assembled structures that are appropriate to transport the prey. In particular, the use of the gripper element to establish a connection in order to move the prey is rewarded.

The quality measure q_i of test t_i can be computed based on the following data:

- $D_i = \max(0, D_i^0 - D_i^{20})$, where D_i^t is the Euclidean distance between the prey and the beacon in test t_i after t seconds of simulation time.
- $d_i^t(j), t \in \{10, 11, 12, \ldots, 20\}, j \in \{1, \ldots, n\} : d_i^t(j) = \max_{a=0}^{35} \mathrm{PP}[i, j, t, \frac{a}{36}2\pi]$, where $\mathrm{PP}[i, j, t, \alpha]$ is the value given in test t_i at time t by the sensor reading of the prey proximity sensor of s-bot j scanning the horizontal direction specified by angle α. $d_i^t(j)$ can be determined by s-bot j at time t.
- Let $M_i^t \subseteq \{1, 2, 3, \ldots, n\}$ be a set of minimal cardinality with the following properties:

[5] That is also the horizontal direction to the center of mass of the prey.

- s-bot j has gripped the prey at time t in test $t_i \Rightarrow j \in M_i^t$,
- $j \in M_i^t \wedge$ s-bot k has gripped s-bot j at time t in test $t_i \Rightarrow k \in M_i^t$.

M_i^t is the set of s-bots that are directly or indirectly connected to the prey. These s-bots are structured in a set of acyclic graphs. Under certain assumptions each s-bot j engaged in test t_i could detect at time t whether ($j \in M_i^t$) is true or false by solely exploiting local information. A real s-bots could use its LED ring to spread its status of being a member of M_i^t to other s-bots that are connected to it. One limitation is that the distribution of the signal would last for some time.

The structure measure s_i of test t_i is given by

$$s_i = \frac{1}{11n} \sum_{t=10}^{20} \sum_{j=1}^{n} s_i^t(j), \tag{3.1}$$

$s_i^t(j)$ is defined by

$$s_i^t(j) = \begin{cases} 1 & \text{if } j \in M_i^t; \\ 0 & \text{if } d_i^t(j) > 50 \wedge j \notin M_i^t; \\ 0.75 & \text{if } d_i^t(j) < 25 \wedge j \notin M_i^t; \\ \frac{50 - d_i^t(j)}{25} * 0.65 + 0.1 & \text{otherwise.} \end{cases} \tag{3.2}$$

Finally, the quality measure q_i is defined as

$$q_i = \begin{cases} s_i & \text{if } D_i = 0; \\ 1 + (R + \sqrt{D_i}){s_i}^2 & \text{otherwise,} \end{cases} \tag{3.3}$$

where $R = 1$ is a constant reward.

In case the prey item has not been moved in test t_i, the quality measure q_i is given by the structure measure $s_i \in [0, 1]$. Otherwise, the fitness is at least 1. In this case the structure measure again has an important influence. Therefore, it is very likely that successfully evolved controllers let the s-bots remain in the vicinity of the prey and to establish physical connection to retrieve the prey.

The term $R + \sqrt{D_i}$ is composed of $R = 1$, a small, distance independent reward, and the root of the distance D_i. Depending on the weight, D_i should not exceed a distance of 100 to 170 cm (see next section). The root function is applied as the scaling function in order to emphasize small differences for lower distance values and to weaken differences for higher distance values.

3.4. Results

In this sections the results concerning three series of experiments are presented. The first part aims at testing a particular aspect of the simulation model, while the following two parts address the problem of cooperative transport introduced previously (see Section 3.2):

- In Section 3.4.1 the skills and limitations of the new controllable *prey proximity sensor* are briefly tested in the context of a discrimination task. A single s-bot is placed in an environment with a cuboid and a cylindrical object in its vicinity. The task is to approach the cuboid and to avoid the cylinder. The s-bot is expected to control the new sensor in order to distinguish the shapes of the two objects.
- Section 3.4.2 presents the results concerning a set of ten evolutionary runs aiming to solve the problem of controlling a group of four s-bots in order to transport small prey of particular shape and different masses.
- Section 3.4.3 presents the results concerning a set of ten evolutionary runs aiming to solve the problem of controlling a group of four s-bots to transport prey of different shapes, sizes and masses.

3.4.1. A Discrimination Test

In order to briefly survey the skills and limitations of the new controllable *prey proximity sensor* introduced previously, the following experiment has been performed.

A single s-bot is placed in the center of the environment with random orientation. Two static prey objects oriented randomly are placed in a distance of 35 cm from the s-bot (one prey on the opposite side than the other with respect to the s-bot location). One prey object is a cuboid of height 10 and the side lengths b_1 and b_2. b_1 and b_2 are uniformly distributed independent random variables in the range of 10 to 16 cm. The other prey object is a cylinder of height 10; its diameter is a uniformly distributed random variable also in range 10 to 16 cm.

The s-bot perceives the current horizontal angle of the prey proximity sensor (i.e., the horizontal direction to which the sensors is pointing to) as well as the reading of the prey proximity sensor. The actuators available are the two active wheels and the horizontal angle for the orientation of the prey proximity sensor that has to be specified. An Elman network with 3 hidden nodes is used as control architecture (see Section 3.3.1).

The s-bot's task is to approach the cuboid within 20 seconds of simulation time. The task requires the s-bot to be capable of disambiguating the cuboid from the cylinder through its prey proximity sensor. This task is particularly hard, since the

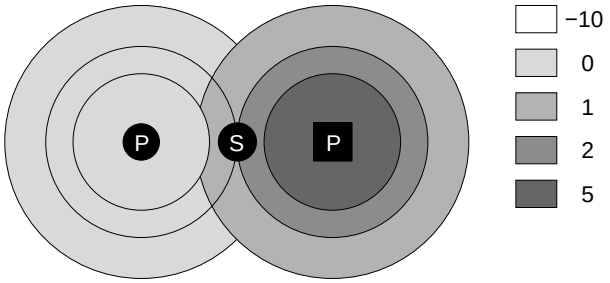

Figure 3.6.: The s-bot (marked by S) is initially placed in the center. Two prey objects (marked by P) - a cylinder and a cuboid - are placed on the left and on the right, each 35 cm away. The task is to reach a position close to the cuboid within the simulation period of 20 seconds. If the s-bot's final position is not more than 25 cm away from the center of the cuboid, the controller gets a maximum reward of 5. If at the end of the simulation period no prey objects is within the sensing range of 50 cm, the controller is punished with a fitness value of -10. Otherwise, if the s-bot has *selected* the wrong item, or if the correct item is not in sensing range the fitness score is 0. Otherwise the fitness score is 1 or 2 depending on the position, as can be seen in the figure.

prey objects are rotated randomly, and the size of each one is varying randomly for each test (up to a factor of 1.6). Therefore, multiple sensor readings are necessary to distinguish the cuboid from the cylinder.

The fitness function is illustrated and explained in Figure 3.6. A sampling technique with sampling size 10 is used. The final fitness score is computed using the normal average of the sample's values.

The fitness of an optimal solution is 5.0. For an s-bot that employs a strategy by which it cannot distinguish the cuboid from the cylinder, the expected fitness score is at most 2.5.

Ten evolutions have been carried out using a population size of 80 ($\mu = 20, \lambda = 60$). The fitness curve is illustrated in Figure 3.7: already after 50 generations some evolved solutions are able to distinguish the cuboid from the cylinder, even if they might fail on some particular problem instances or because of the noise affecting sensors and actuators. Due to the noisy fitness estimation, the fitness values of the best rated individuals tend to be overestimated (see Section 2.3.2). The average value of the re-evaluated 20 parent individuals is not biased, since the re-evaluation takes place after selection is applied. It can serve as an approximation for a lower bound of the fitness of the best individual.

The symmetric problem *to move to the cylinder within 20 seconds of simulation time*, seems to be more difficult than the problem to move to the object that is *not* the cylinder: Figure 3.8 presents the development of fitness values for 10 evolutions concerning the analog task. It seems that only in a few runs controllers have been evolved that are able to solve the task with a quite low failure rate.

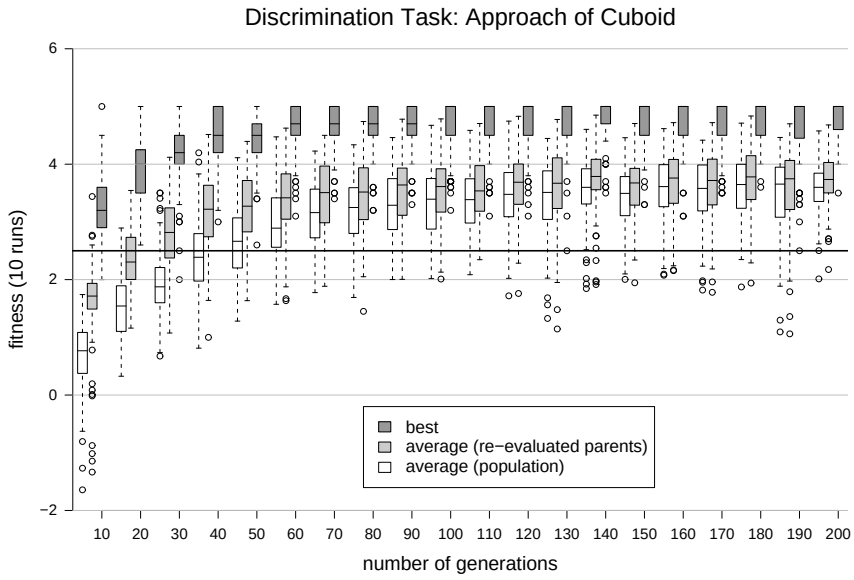

Figure 3.7.: The fitness of an optimal solution is 5.0. For a solution that is not able to distinguish the cuboid from the cylinder, the expected fitness score is at most 2.5. This bound is marked by the horizontal line. For example, a strategy that always moves close to the first prey that is scanned by the controllable prey proximity sensor, has an expected fitness score of 2.5.

Altogether, the results of the evolutions concerning the task to approach the cuboid outperform the results obtained concerning the symmetric problem. We believe that the reason for this is that a cuboid have properties (the corners) that are easier to be detected. Therefore, evolved solutions tend to rest at places near a corner. Since the round shape of a cylinder looks similar to the flat side of cuboid, it might be more difficult to identify cylinders in systems where sensor readings are highly affected by noise.

However, it might be that a potential solution in which the s-bot precisely detects the object with a corner and then moves towards the other object, has not evolved due to the time-limits imposed on the task.

3.4.2. Transport of Prey of a Particular Shape and Different Masses

In this section, we present the results concerning the evolution and analysis of transport behaviors of groups of s-bots moving small prey objects of particular shape and different masses.

A set of 10 independent evolutionary runs has been performed for 750 generations.

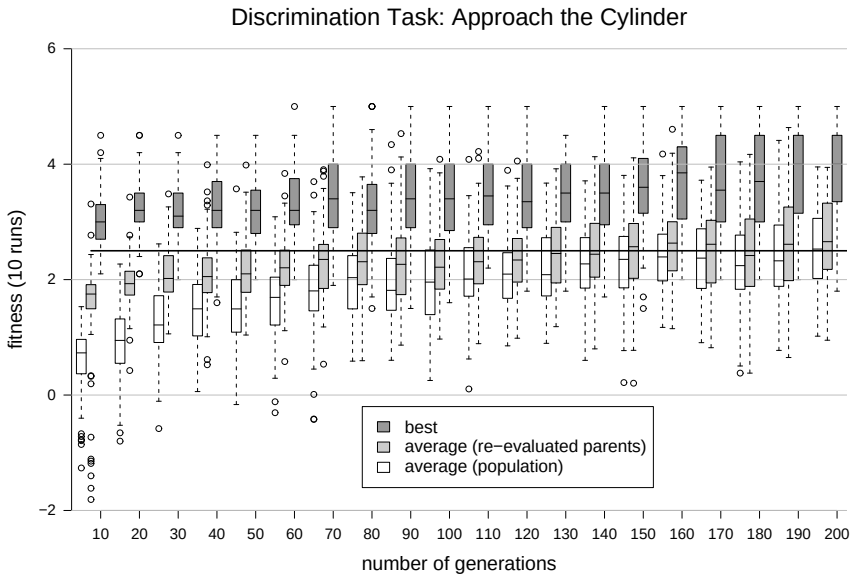

Figure 3.8.: Ten evolutions have been performed in order to solve the symmetric problem - to let the s-bot approach the cylinder instead of the cuboid. The fitness of an optimal solution is 5.0. For a solution that is not able to distinguish the cuboid from the cylinder, the expected fitness score is at most 2.5. This bound is marked by the horizontal line. A strategy that always moves close to the first prey that is scanned by the controllable prey proximity sensor, has an expected fitness score of 2.5.

One single run lasts 3-4 weeks of computational time.[6]

The samples that have been used to evaluate individuals during evolution are composed of 5 tests each. In each test, the prey is modeled as a cylinder of radius 8 cm and height 12 cm. The masses of the prey in the five test of each sample are 500, 625, 750, 875 or 1000 g.

In the following, some features of the evolved solutions, such as the transport performance, the obtained behaviors, the flexibility, and the ability to scale using larger group sizes are analyzed. In Section 3.4.2.1 the performances of evolved controllers are analyzed for every evolutionary run. Focus is given on a measure reflecting the distance the prey has gained with respect to the target. Well performing individuals are selected to be analyzed in the subsequent sections. In addition, the robustness of the s-bot groups with respect to a different initial positioning method is analyzed. In Section 3.4.2.2 the behaviors obtained at the level of an s-bot and at the group level are described. The ability of the group of s-bots to adapt to environmental changes is

[6]Machines equipped with 512 MB of memory and a 1,537 MHz processor (AMD Athlon XP™ 1800+) are utilized.

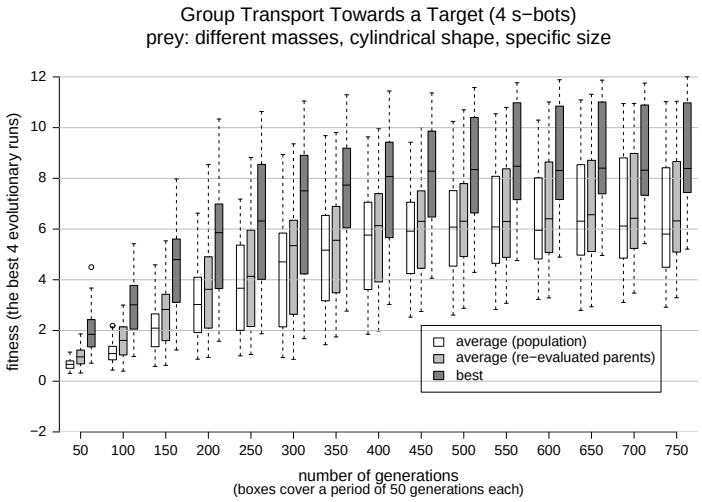

Figure 3.9.: Box-and-whisker plot providing information about the evolutionary progress concerning the 4 most successful evolutionary runs (total number of runs: 10). Characteristics about three types of observations are displayed (partitioned into groups of 50 generations): the best fitness value of a population (dark gray boxes), the average fitness value of all individuals of the population (white boxes), and the average fitness value of the re-evaluated parent individuals (light gray boxes).

evaluated in Section 3.4.2.3. The group engaged in the transport task shall adapt the direction of movement according to a new target that appears dynamically. Finally, in Section 3.4.2.4 the ability of the evolved individuals to control groups of increased size in order to transport prey objects of increased mass is assessed.

3.4.2.1. Performance and Robustness

Among the ten evolutionary runs the fitness level, attained by the *best* evolved individuals in each case, is varying considerably ranging from very poor to remarkably high fitness values. In this section, for every evolutionary run some individuals are selected for a performance analysis. For four particular evolutions, a well performing individual is selected to be analyzed in the subsequent sections. The four *most successful* evolutionary runs and the six *less successful* ones are illustrated in Figure 3.9 and Figure 3.10 respectively. Both figures show the development of the best fitness (dark gray boxes), average fitness of all individuals of a population (white boxes), and the average fitness of re-evaluated parent individuals (light gray boxes).

Since the fitness evaluation procedure is affected by noise and since the evaluations are based on varying initial conditions, individuals exhibit performance fluctuations. Thus, the best fitness values that refer to the best rated individuals in the particular generations are very likely to be over-estimated. This over-estimation might be

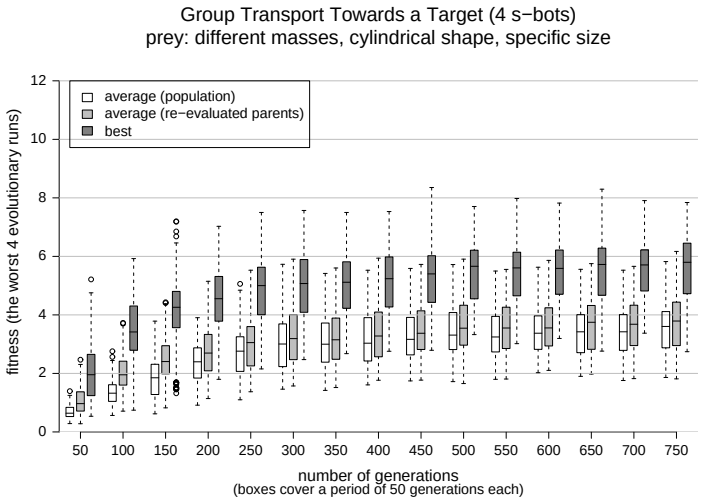

Figure 3.10.: Box-and-whisker plot providing information about the evolutionary progress concerning the 6 least successful evolutionary runs (total number of runs: 10). Characteristics about three types of observations are displayed (partitioned into groups of 50 generations): the best fitness value of a population (dark gray boxes), the average fitness value of all the individuals of the population (white boxes), and the average fitness value of the re-evaluated parent individuals (light gray boxes).

typical, if large populations are used or if a considerably large amount of individuals can get comparatively high fitness values. However, since the μ parent individuals that are copied into the next generation due to the elitism of the plus-strategy used get re-evaluated, there is at least no tendency to a systematic over-estimation due to previous fitness evaluations.

The average of the re-evaluated fitness values may fluctuate, but is not biased since the re-evaluation takes place after selection is applied. Therefore, it can serve as a *vague lower bound* for the quality level reached.

It can be seen that in average there is a continuous, sublinear increase of fitness values for the entire range of 750 generations. We suppose that the average quality of result can be improved by increasing the total number of generations.

Each fitness value is computed as the average of the observed quality measures q_1, q_2, q_3, q_4 and q_5 concerning five tests against a sample of different conditions. The quality measure accounts for the transport distance as well as for the group structure (see Equations 3.1, 3.2 and 3.3 in Section 3.3.3). In the following analysis we focus on a measure related to the transport distance.

The new measure to evaluate the performance of a solution exhibited in a test is the *distance gain*. The distance gain G_i in a test t_i is defined by

Table 3.3.: Distance gain (within 20 seconds) using a perfectly aligned, pulling chain (of pre-connected s-bots) that is initially attached to the prey. The root function is used for scaling purpose (see Equation 3.4). Here, any noise acting on sensor or actuators is completely disabled.

	500 g	625 g	750 g	875 g	1000 g
2 s-bots	11.99	10.21	8.03	4.99	0
3 s-bots	13.99	13.00	11.93	10.75	9.426
4 s-bots	14.94	14.26	13.54	12.78	11.98

$$G_i = \begin{cases} \sqrt{D_i^0 - D_i^{20}} & \text{if } D_i^0 \geq D_i^{20}; \\ -\sqrt{D_i^{20} - D_i^0} & \text{otherwise,} \end{cases} \tag{3.4}$$

where D_i^t denotes the Euclidean distance (in cm) between the prey and the beacon in test t_i after t seconds of simulation time. Therefore, G_i is the distance gained concerning the prey object's transport (with respect to the beacon) during the entire simulation period in test t_i scaled by a root function. If the prey gets closer to the beacon, the distance gain is positive. If the prey is moved away from the beacon, the distance gain is negative. The root function is applied as the scaling function in order to emphasize small differences for lower distance values and to weaken differences for higher distance values.

Table 3.3 lists the observed distance gain corresponding to a setup in which a group of s-bots, structured in a connected, linearly aligned chain, is pulling the prey item for a period of 20 seconds towards the beacon. The first s-bot of the chain is connected to the prey right from the start. All s-bots are controlled by a handwritten controller in order to move backwards applying the maximum angular speed to the wheels. Here, there is no noise affecting the wheels, and all wheels behave identically, to ensure that the chain remains in its linear structure. Several combinations of chain sizes and prey masses are tested to gives us an idea about the potential distance gain within 20 seconds.

However, we expect that the distance gain values obtained by the evolved controllers will result lower than the distance gain values listed in Table 3.3, since the s-bots will be evaluated starting disassembled from random locations up to half a meter away from the prey.

Post-evaluation of Individuals' Performances As done in the analysis of the previous experiments (see Section 2.4), the set of $\mu = 20$ parent individuals of each final generation are re-evaluated on a fixed and common sample of 500 different tests[7]

[7]For each particular mass of the prey (i.e., 500, 625, 750, 875 and 1,000 g) 100 tests are performed.

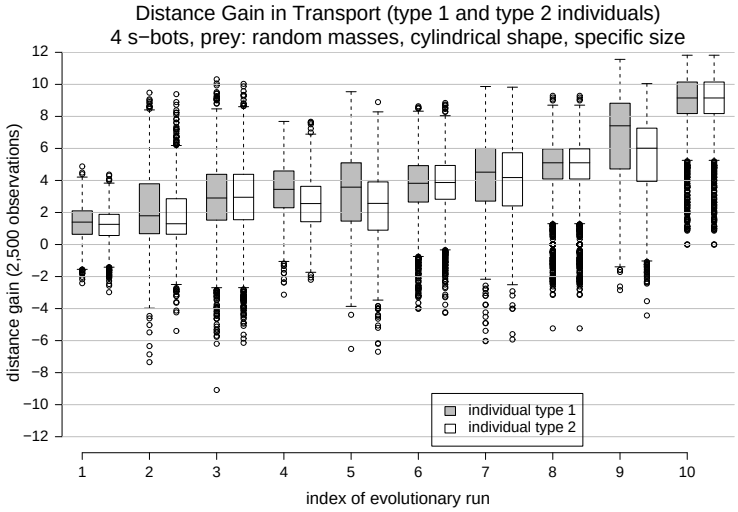

Figure 3.11.: Distance gain in transport by the individuals of type 1 and type 2 of each evolutionary run. The distance gain is a measure of the total change of distance (in cm) between the prey and the target (the beacon) during the 20 seconds of simulation time. The root function is used for scaling purpose (see Equation 3.4). Positive values indicate that the target has been approached.

to compute the quality measures as defined by the experimental setup (see Equation 3.3 in Section 3.3.3). For each evolutionary run, the individual with the highest observed average performance (also called *type 1 individual*) and the one with the lowest observed standard deviation (also called *type 2 individual*) are selected.

The type 1 and the type 2 individual are post-evaluated for a second time on a random, but fixed set of 2,500 tests.

Figure 3.11 presents the distance gain observed in these 2,500 tests for both types of best individuals, for all evolutionary runs (ordered by the median performance of the type 1 individuals). It can be seen that some solutions do not always move the prey towards the beacon. However, this happens only occasionally: for each of all type 1 and type 2 individuals the fraction of observations having a distance gain of less than -2 is less than 3.5%.[8] The type 1 individual of the evolutionary run number 10 exhibits a remarkable performance: in more than 90% of the tests, the distance gained is between 50 and 140 cm and the distance gain is never negative.

The mass of the prey should have a high influence on the distance that is gained.[9]

[8]This corresponds to a movement of the prey of at least 4 cm away from the beacon.

[9]Note that a) the force necessary to overcome the frictional forces increases with the mass of the prey, b) the angular speed and the force driving the s-bot's wheels are limited, c) the wheels might slide, if an object is pulled or pushed, and d) acceleration is lower for objects that are heavy when applying a constant force.

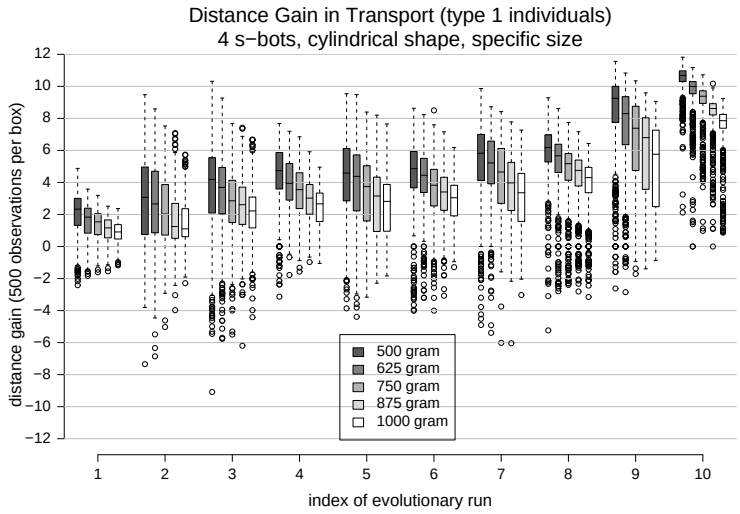

Figure 3.12.: Distance gain in transport by type 1 individuals concerning different prey masses.

Transporting heavy prey objects might require a more complex coordination among the s-bots in order to be performed efficiently.

Figure 3.12 illustrates the performance deviations with respect to different masses of prey. In general the individuals of run 1 and 2 exhibit very weak performances. For the type 1 individuals of runs 3-6 the median values of the distances that the lightest prey is moved towards the beacon are 17.55 cm, 22.45 cm, 21.00 cm and 23.62 cm. For runs 7-10 the corresponding median values are 33.93 cm, 38.22 cm, 85.67 cm and 104.08 cm. In addition, the latter runs (especially runs 9 and 10) produced individuals that exhibit an acceptable performance concerning prey masses up to 1,000 g. The median values of the distance gain concerning all observations of the type 1 individuals of run 9 and 10 are 54.95 cm and 83.72 cm.

Obviously, when looking at the observations corresponding to a particular mass, the performance fluctuations exhibited in the 500 tests by the type 1 individual of run 10 are remarkably low. The standard deviations of the observations concerning mass 500, 625, 750, 875 and 1,000 g are 1.04, 1.51, 1.44, 1.46 and 1.64.

In the remaining part of this chapter, we focus on the analysis of the four type 1 individuals from the evolutionary runs 7, 8, 9 and 10.

Transport Performance Concerning a Different Initial Positioning In some of the following analysis an alternative positioning method concerning the s-bots' initial placement is utilized (see Figure 3.13). This alternative method is referred to as the

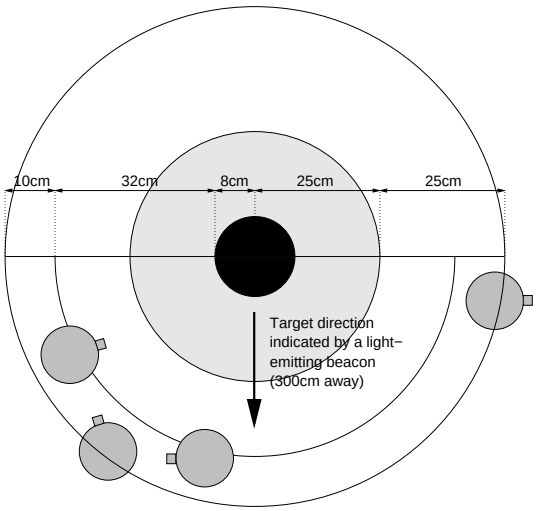

Figure 3.13.: Example of initial placement: the prey object (black) is a cylinder with radius 8 cm. It is placed in the center and it has to be moved in the direction of a light emitting beacon (downward, as indicated by the arrow). According to our original positioning method, four s-bots (gray) are placed at random positions within a semi-circle, at least 40 cm and at most 50 cm away from the center. The alternative positioning method places the s-bots to random locations within the full circle but at least 25 cm and at most 50 cm away from the center. In both positioning methods, the s-bots are oriented to random directions.

full-circle positioning. It is similar to the one used in the experiments of the previous chapter, and it allows to place s-bots up to 50 cm away from the center of prey in all horizontal directions. The minimal distance between the initial location and the center of prey is 25 cm. During the evolutions, the s-bots have been placed 40 to 50 cm away from the center of prey to emphasize the evolution of s-bot controllers that are able to approach the prey once placed in the vicinity. Moreover, the s-bots have been placed within a semi-circle that is oriented towards the beacon. Using the *full-circle positioning* method enables us to investigate the groups of sizes up to 20 s-bots without the use of an additional recruitment mechanism.

Changing the initial placement of s-bots might diminish the group's performance. The four type 1 individuals of the evolutionary runs 7-10 are evaluated on 1,250 tests using the full-circle positioning method. The results are shown in Figure 3.14. The white boxes correspond to the original initial placement. The (light and dark) gray boxes refer to the full-circle positioning.

By comparing the white and light gray boxes it can be seen that the new placement method results in a performance decrease for all four individuals. However, in case of the individuals of runs 9 and 10, the attained performance levels are still acceptable.

In both considered cases the tests last 20 seconds each. If we extend the simula-

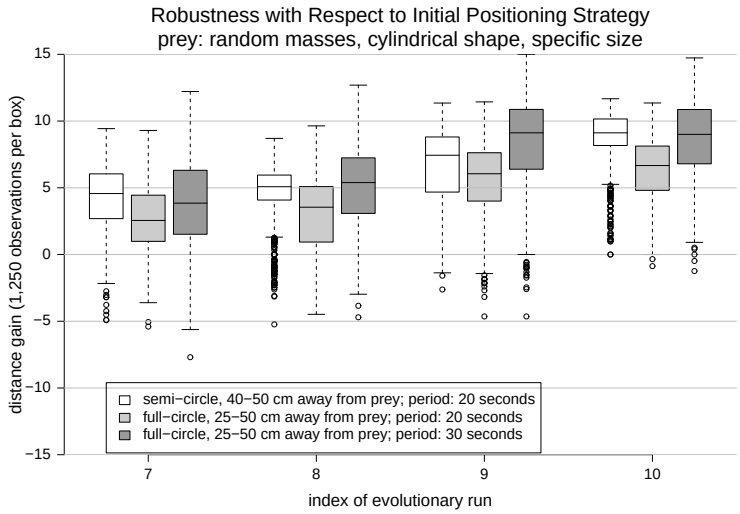

Figure 3.14.: Test to what extend the s-bot groups are robust with respect to two different initial placement methods. Each white box refer to 1,250 observations concerning the original initial placement that is used during the evolutionary runs. The other boxes refer to observations (1,250 each) for the *full-circle positioning* method.

tion period for additional 10 seconds, the performances of the individuals increase again. Unfortunately, the availability of additional time does not reduce the increased amount of performance fluctuations (as indicated by the dark grey boxes and the whiskers) of the individuals of run 8 and 10.

3.4.2.2. Behavioral Analysis

In the following, certain aspects concerning the behaviors of an s-bot or a group of s-bots are analyzed.

Individual Behavior In order to gain insight into the mode of operation of an s-bot controlled by an evolved individual, certain variables, such as the left and right wheel's speed, have been monitored during the whole simulation period of 20 seconds. The obtained data concerning an s-bot controlled by the best performing evolved controller individual (i.e., the type 1 individual of run 10) is visualized in Figure 3.15.

For the whole period of simulation, the right wheel is set to a desired velocity of around $8s^{-1}$. The left wheel and the rotational degree of the turret are used to adjust the direction of motion. At the beginning the s-bot turns on the spot and the turret is rotated for approximately 180 degrees with respect to the chassis. The s-bot approaches the prey moving backward; however, the gripper is heading in the direction of movement. Once a connection to a prey or to another s-bot has been

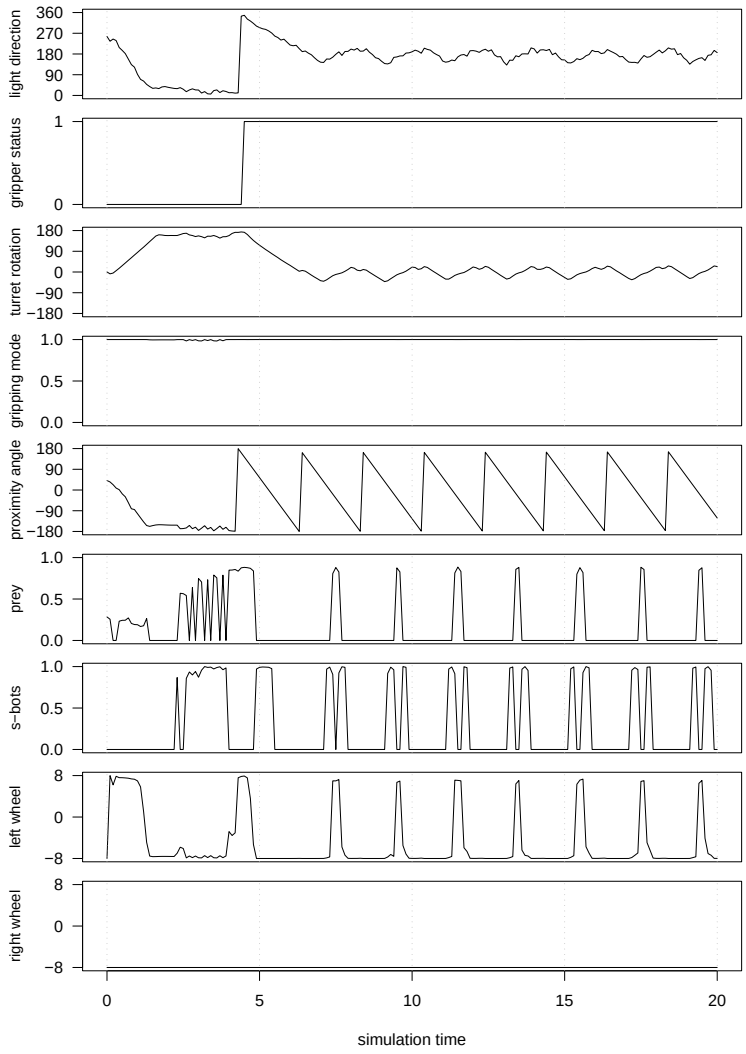

Figure 3.15.: Certain variables have been monitored during the life-time of one particular s-bot engaged in group transport: a) the direction of light perceived (with respect to the chassis), b) the status of the gripper element, c) the current angular offset between the turret and the chassis, d) the activation of the neural network output used to set the gripping mode, e) the angular shift that is applied to the orientation of the prey/s-bot proximity sensor, f) the prey proximity, g) the s-bot proximity, and h) the desired velocities of the left and right wheels.

established by the gripper element, the chassis is once again rotated for 180 degrees in order to start pulling the gripped object. It can be seen that the gripper element is always in gripping mode. After 4.5 seconds the s-bot has connected to another object.

According to the experimental setup the prey and s-bot proximity sensor is heading to the prey for the first second of simulation time (see Section 3.3.3). For the remaining 19 seconds it is controlled by the angular shift that is plotted in Figure 3.15. As long as the gripper has not established any connection, the s-bot tries to keep or get contact to the prey. In this period, the prey sensor is aligned in the direction of the gripper element.[10] As soon as the s-bot has connected to another object, the sensor starts circulating around the s-bot perceiving two s-bots and one prey with a frequence of $0.5s^{-1}$. It seems that the perceived proximity of the prey affects the speed of the left wheel.

Group Behavior The fitness function used in the evolutionary algorithm rewards the presence of assembled s-bot structures connected to the prey (see Section 3.3.3). To get a maximum reward, each s-bot should establish a connection either directly to the prey or to another s-bot that is directly/indirectly connected to the prey. Given a group of four s-bots there are 17 possible structural configurations of s-bots connected both to each other and to the prey.[11] These structures are illustrated in Figure 3.16. Note that in this illustration the concrete locations of the connection points are chosen arbitrarily.

For each configuration in the figure the frequency of occurrence and the corresponding median distance gain concerning 2,500 tests of the type 1 individual of run 10 are displayed. Note that the observed structures are recorded at the end of the simulation, while the distance gain is the result of the group behavior during the entire simulation.

16 out of 17 structures have appeared for the individual of run 10; five structural configurations among them appeared less than 10 times. In 46.88% of the cases an s-bot chain of length two as well as two solitary s-bots are connected to the prey. In 27.20% of the observations all s-bots are directly connected to the prey. In 11.51% of the cases less than 4 s-bots are part of the structure (left side in the figure).

Figure 3.17 shows the frequencies and the distance gains corresponding to various structures that have been observed during 2,500 tests of the type 1 individual of run 9. For this individual all structures have appeared; however, some with very low frequency. Again the case that all s-bots are directly connected to the prey occurs the most often (25.60%). There are numerous of other well performing structures

[10] The sensor is set to an angle of around 180 degree with respect to the chassis. The angular offset of the turret (including the gripper element) and the chassis is also around 180 degree.

[11] Structures that are isomorphic count only once.

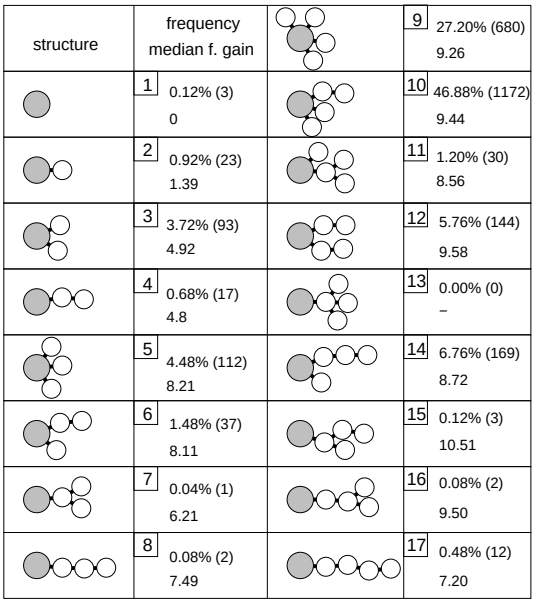

Figure 3.16.: List of possible structural configurations of four s-bots (directly or indirectly) connected both to each other and the prey. Concerning 2,500 tests for the type 1 individual of evolutionary run 10, the frequency and the median distance for every particular structure is presented.

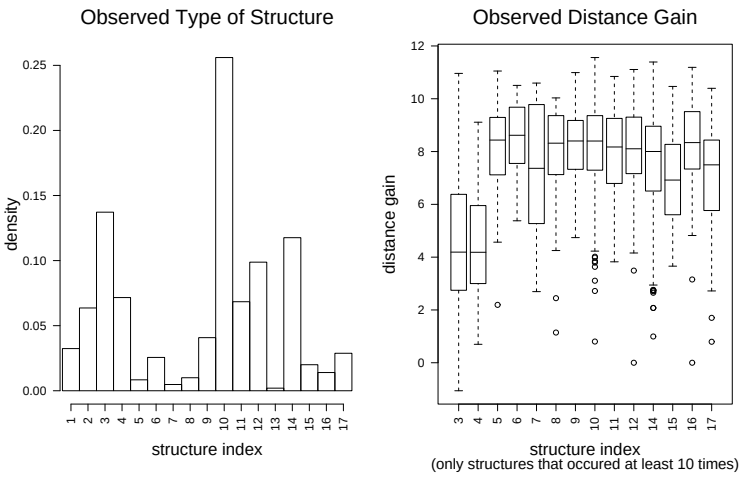

Figure 3.17.: Frequency and distance gain of structures observed during 2,500 tests with the type 1 individual of evolutionary run 9. The structure indices correspond to the structures shown in Figure 3.16.

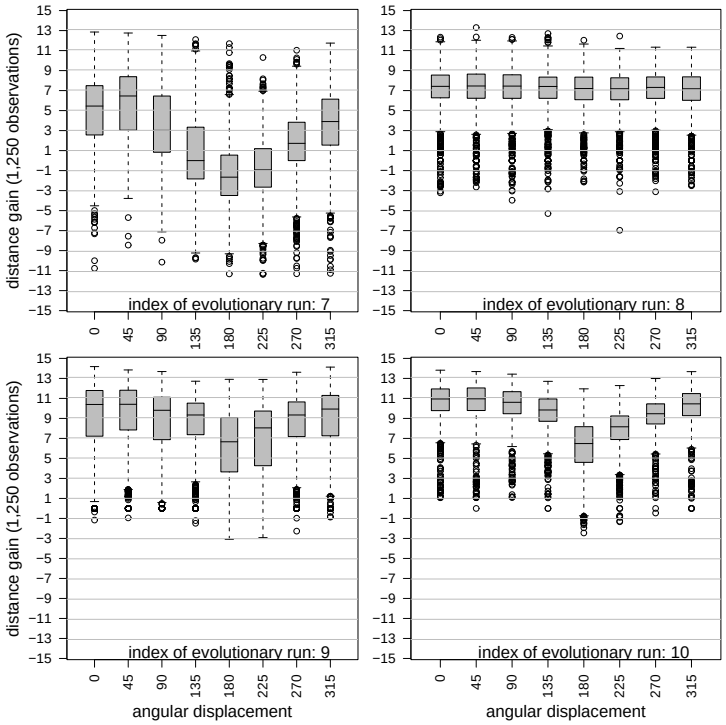

Figure 3.18.: Flexibility test: ability of a group engaged in the transport of a prey to adapt the direction of transport according to a new target. The distance gained within the second part of the simulation and with respect to the new target is illustrated (scaled by root functions). Eight different angular displacements between the old and the new target are considered.

(also some of only three s-bots) that occur frequently.

3.4.2.3. Flexibility

In the experiments described in this chapter, the simulated environment is equipped with a single, light emitting beacon indicating the desired direction of transport. The beacon is placed 300 cm away from the prey. The neural network controllers synthesized within the evolutions let the group of s-bots form a structure in order to transport the prey into the desired direction.

In this section we analyze to what extend a group of s-bots, engaged in the transport of a prey, is able to adapt dynamically the direction of transport according to a new target.

The simulation period of each test is doubled from 20 to 40 seconds. Concerning the first half of the simulation period, the basic experimental setup has been retained

unchanged: the s-bots are placed within a semi-circle around the prey and the beacon is located 300 cm away from the prey (for details see Section 3.3.3).

As soon as the first half of the simulation period is elapsed, the beacon (i.e., the target) is placed to a new position fulfilling the following two conditions:

- The new position has the same height and it is 300 cm away from the current location of the prey.
- The (horizontal) angular displacement of the beacon with respect to the prey can be 0, 45, 90, 135, 180, 225, 270 or 315 degrees.

For each angular displacement the type 1 individuals of the runs 7, 8, 9 and 10 have been evaluated on 1,250 tests. As can be seen in Figure 3.18 one s-bot controller individual acts robust with respect to angular displacements of all magnitudes, while the performance of others decreases especially in case the beacon is placed in the opposite direction (180 degrees). It is interesting that the decrease of performance is not fully symmetric concerning angular displacements to the left or to the right.

The individual of run 7 performs weak in case the angular displacement to the left or to the right exceeds 90 degrees. For the other three controllers, the median performances concerning all tested angles of displacement correspond to transport distances of at least 42 cm. We conclude that three out of four controllers are able to respond efficiently to a dynamic repositioning of the target.

3.4.2.4. Scalability

In this chapter, so far we have considered the case that the group that is engaged in the transport task is composed of four s-bots. We have evolved controllers that let such a group transport prey items from 500 to 1,000 g. Objects of 500 g require the coordination of 2 or more s-bots to be moved efficiently. Most of the evolved individuals are able to let a group of 4 s-bots move a prey of mass 500 g with an acceptable performance. In case of an object of weight 1,000 g four s-bots need to coordinate properly in order to transport the prey in an efficient way, and still benefit can be gained by using more than 4 s-bots.

In this section we analyze the controllers' abilities to scale from group size 4 to the group sizes 8, 12 and 16. The mass of the prey is also modified linearly from 500 to 1,000, 1,500 and 2,000 g.

For each individual of the evolutionary runs 7-10 and for each group size 250 tests have been performed. The simulation period has been extended to 30 seconds, since we expect that larger groups of s-bots will require more time in order to form a desired structure. Since recruitment techniques are not taken into account, we have to ensure a proper placement of groups up to 16 s-bots in the immediate vicinity of the prey. The prey has to be located within the rather small sensing range of 50 cm

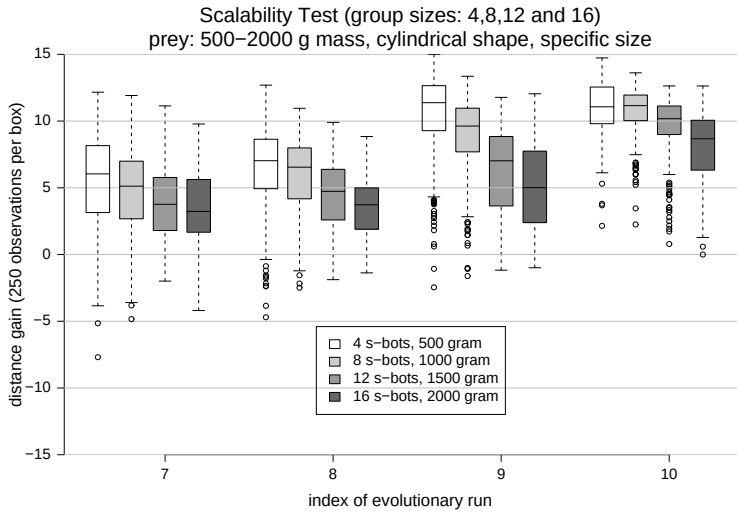

Figure 3.19.: Scalability test: distance gain within 30 seconds of simulation time with groups of size 4, 8, 12 and 16 transporting prey objects of mass 500, 1,000, 1,500 and 2,000 g.

of each s-bot. Therefore, the full-circle positioning method described and analyzed in Section 3.4.2.1 is utilized. Compared to the default placement strategy, this method causes a performance decrease in our systems with group size 4 in case the mass of the prey is chosen randomly from 500, 625, 750, 875 or 1,000 g. However, we have observed that the type 1 individuals of the runs 8, 9 and 10 are at least robust enough to reach an acceptable performance level, especially in case the simulation period is increased.

Figure 3.19 presents the results concerning 250 tests per individual and for each group size. Overall, the observed scaling abilities are not satisfying. It seems to be unfeasible for the evolved individuals to let the s-bots transport prey of arbitrary high masses by adapting the group size linearly with respect to the mass.

However, it can be seen that for the particular group sizes 4, 8, 12 and 16 the evolved individuals attain an acceptable performance level: the individual type 1 of run 10 is able to transport the prey objects with a group size 8 to 12 without loosing much efficiency. Also in case 16 s-bots are used the performance is still fairly good: in 75% of the cases the prey is moved towards the beacon for more than 40 cm. In half of the observations, the distances gained are more than 75 cm. Also the performance of the type 1 individual of run 9 is acceptable for all group sizes. Only the performance of the individual of run 8 concerning group size 16, as well as the performance of individual 7 concerning the group sizes 12 and 16 are not satisfying.

We suppose that the performance decrease in case of larger group sizes is mainly caused by the increased fraction of s-bots that do not connect directly or indirectly

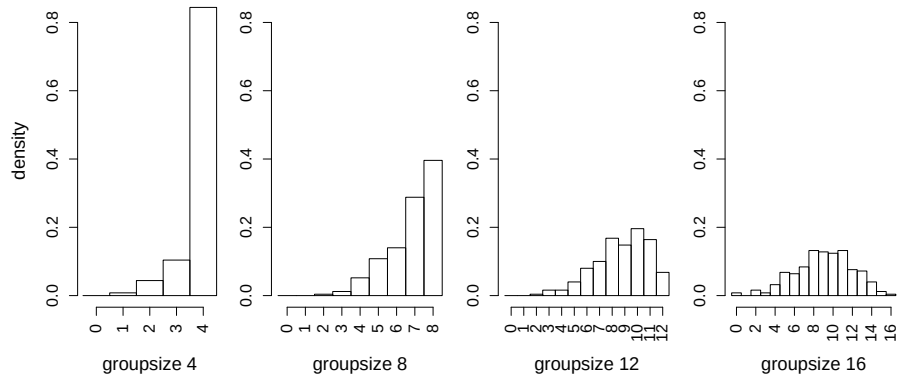

Figure 3.20.: Frequency of occurrence of different structure sizes: for the 1,250 tests in which the type 1 individual of run 10 controls a group of 4, 8, 12 or 16 s-bots in order to transport a prey of mass 500 to 2,000 g, the structure size, i.e., the number of s-bot directly or indirectly connected to the prey object has been recorded. Here, the frequencies that have been observed for each group size are given.

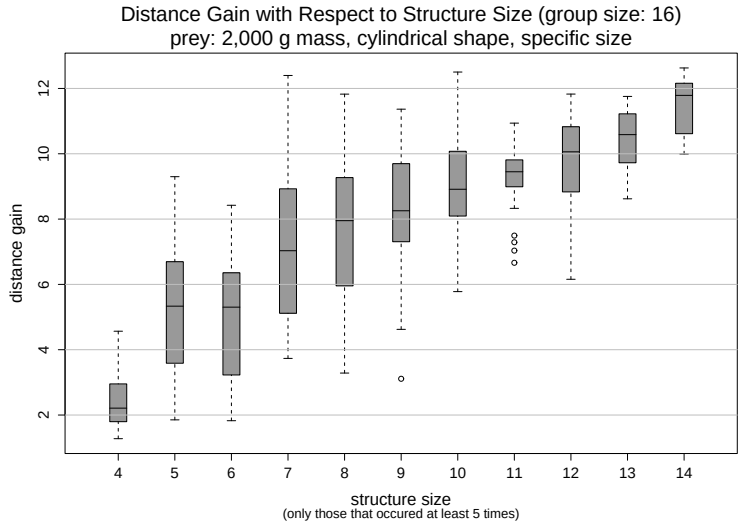

Figure 3.21.: Distance gain for different structure sizes. For the 750 tests of the type 1 individual of run 10 the distance gain is plotted against the size of the corresponding assembled transport structure observed at the final time step of the each test.

to the prey. Concerning the scalability tests of this section, Figure 3.20 shows the distribution of the number of s-bots that are part of the (connected) transport structure for different group sizes at the end of each test. The skewness of the distribution observed for group size four tends to dissappear with increasing group sizes. As we can see in Figure 3.21, structures of increasing size perform better than smaller ones.

In case of the tested combinations of group size and mass the prey can be retrieved even if some s-bots fail to get involved, since there is a redundancy. However, benefit could be gained if the fraction of s-bots actively engaged in the transport would be increased.

We conclude that the performances of all controller individuals decrease in case both group size and the mass of the prey are increased linearly. However, some solutions attain still a level of performance that is acceptable for group sizes of up to 16 s-bots. We expect that these solutions cannot be applied successfully to groups much larger than 20 s-bots.

3.4.3. Transport of Prey of Different Shapes, Sizes and Masses

In this section the results concerning the evolution and analysis of transport behaviors of groups of s-bots moving prey objects of different shapes, sizes and masses are presented.

Ten independent evolutionary runs have been performed for 850 generations. One single run lasts 4 weeks of computational time.[12]

The samples that have been used to evaluate individuals during evolution are composed of 5 tests each. For every sample, the masses of the prey corresponding to the five tests are 500, 625, 750, 875 or 1,000 g. In each test the prey object is modeled either as a cylinder or as a cuboid (see Section 3.1.1), both shapes are selected with equal probability. In addition, the prey's size is chosen randomly (independent from the mass) for each test:

- The height of the prey in each test is set randomly to either 10 or 20 cm, each choice with equal probability. If a tall prey is located between an s-bot and the light emitting beacon that indicates the desired direction of transport this s-bot is not able to perceive the target.

  As in the previous experiments an s-bot that is located in front of a prey object cannot perceive other teammates that are on the opposite side. On the other hand, an s-bot can still detect the prey in case another s-bot is placed in between.[13]

[12] Machines equipped with 512 MB of memory and a 1,537 MHz processor (AMD Athlon XP™1800+) are utilized.

[13] The s-bot's camera system can detect another s-bot as long as its torso is visible. In these experiments, the prey objects are in any case taller then the s-bot's torso.

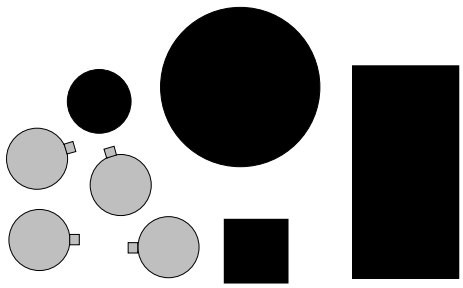

Figure 3.22.: The prey object that has to be transported in a given direction can be a cuboid or a cylinder of different size and mass. In this top-view, prey of the most extreme sizes are visualized (by the black objects) to give an idea about the variation of the sizes among prey objects as well as the proportions with respect to the size of the s-bots (gray objects). The mass of the prey does not depend on its shape or the size. Therefore, the smallest prey can be of biggest mass.

- In case the prey is modeled as a cylinder the radius is chosen randomly according to a uniformly distributed variable within the bounds of 6 and 15 cm.

- In case the prey is modeled as a cuboid, the length l of one horizontal side is chosen according to a uniformly distributed random variable ranging from 12 to 40 cm. The other horizontal side length is set to a uniformly distributed random variable with the lower bound of 12 cm and the upper bound of $\min(l, 20)$ cm. Therefore, only one side of the cuboid can have a length bigger than 20 cm.

Figure 3.22 illustrated the proportions of different prey objects with respect to the s-bot's size.

Most of the following analyses take place in the same way as they have been carried out in Section 3.4.2. Again, the transport performance, the obtained behaviors, the flexibility, and the ability to scale using larger group sizes are discussed. In Section 3.4.3.1 the performances of evolved controllers are analyzed for every evolutionary run. It is evaluated to what extend the performance of the s-bots' group is affected by different shapes, sizes and masses of prey. Promising individuals are selected for further analysis. In addition, the robustness of the s-bot groups with respect to a different initial positioning method is analyzed. In Section 3.4.3.2 the behaviors obtained at the level of an s-bot and at the group level are described. The ability of the group of s-bots to adapt to environmental changes is evaluated in Section 3.4.3.3. The group engaged in the transport task shall adapt the direction of movement according to a new target that appears dynamically. Finally, in Section 3.4.3.4 the ability of the evolved individuals to control groups of increased size in order to transport prey objects of increased mass is assessed.

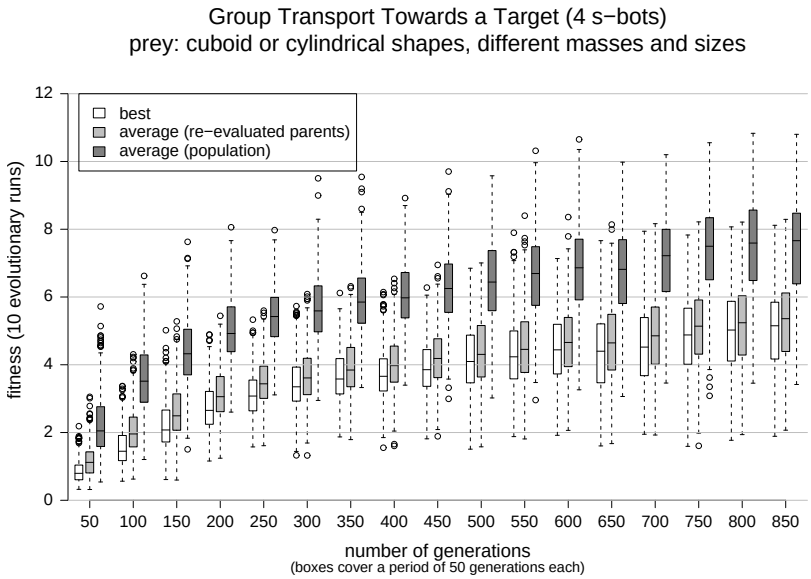

Figure 3.23.: Box-and-whisker plot providing information about the evolutionary progress in 10 evolutionary runs. Characteristics about three types of observations are displayed (partitioned into groups of 50 generations): the best fitness value of a population (black boxes), the average fitness value (white boxes), and the average fitness value of the re-evaluated parent individuals (grey-boxes).

3.4.3.1. Performance and Robustness

In this section, for every evolutionary run some individuals are selected for a performance analysis. For six particular evolutions, well performing individuals are selected for further analysis.

Figure 3.23 shows the development of the best fitness (dark gray boxes), average fitness (white boxes), and average fitness of re-evaluated parent individuals (light gray boxes). It can be seen that in average there is a continuous increase of fitness values for the entire range of 850 generations. We suppose that the average quality of the results can be improved significantly by increasing the total number of generations.

Post-evaluation of Individuals' Performances As done in the analysis of the previous experiments, for each evolutionary run two types of individuals are selected from the final population. Therefore, the set of $\mu = 20$ parent individuals of each final generation are re-evaluated on a fixed and common sample of 500 different tests[14] to compute the quality measures as defined by the experimental setup (see Equation 3.3 in Section 3.3.3).

[14]For each particular mass of the prey (i.e., 500, 625, 750, 875 and 1,000 g) 100 tests are performed.

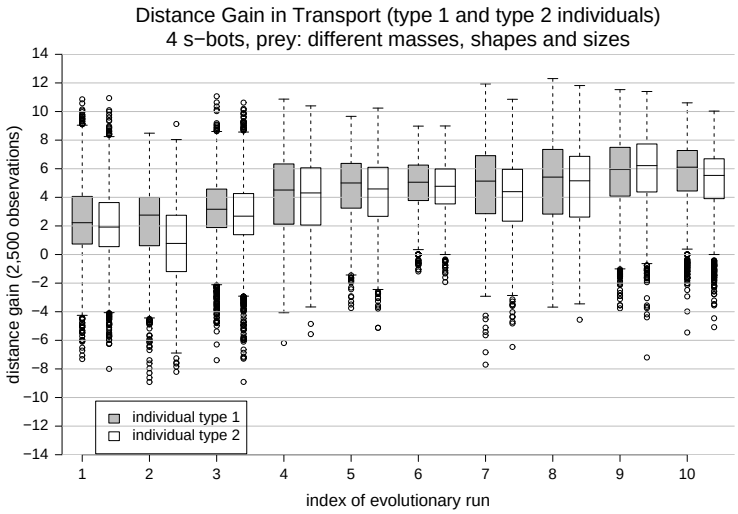

Figure 3.24.: Distance gain in transport by the individuals of type 1 and type 2 of each evolutionary run. The distance gain is a measure of the total change of distance (in cm) between the prey and the target (the beacon) during the 20 seconds of simulation time. The root function is used for scaling purpose (see Equation 3.4). Positive values indicate that the target has been approached.

For each evolutionary run, the individual with the highest observed average performance (the type 1 individual) and the one with the lowest observed standard deviation (the type 2 individual) are selected.

Both types of individuals are post-evaluated for a second time on a random, but fixed set of 2,500 tests. This time, the performance measure used is the distance gain (see Equation 3.4 in Section 3.4.2.1).

Figure 3.24 presents a box-and-whisker plot concerning the distance gain observed in the 2,500 tests for both types of individuals and for all evolutionary runs. The performance deviations with respect to different masses of prey are illustrated in Figure 3.25. The evolutionary runs are ordered from the left to the right by the median performance observed for the type 1 individuals.

In comparison with the performance levels attained by the type 1 and type 2 individuals concerning the evolutionary runs of the previous experiments (see Section 3.4.3.1) no individual achieved a remarkable performance; however, a bigger amount of individuals attained an acceptable level of performance: the fraction of type 1 individuals that are able to let a group of s-bots move the prey in 50% of all observations for 25 cm towards the beacon (i.e., a distance gain of 5) or even more is twice as big as in the previous set of experiments.

The highest median performance is achieved by a type 2 individual, the one with index 9. In this case, the median distance gain corresponds to a distance of 39 cm.

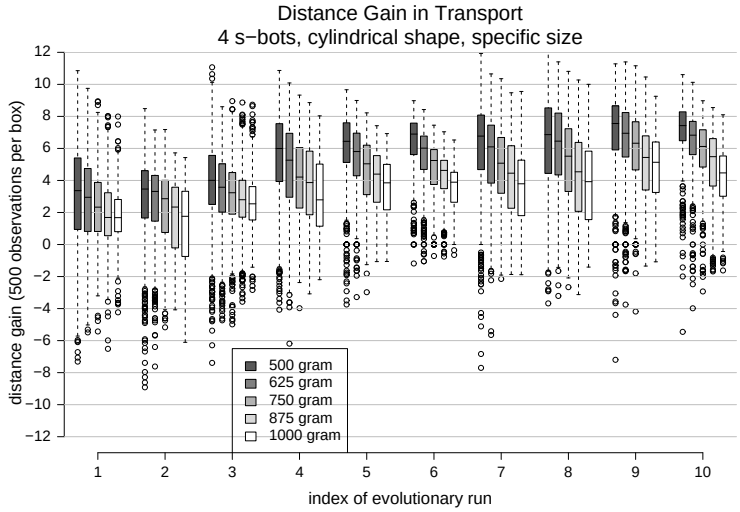

Figure 3.25.: Distance gain in transport for different prey masses. For each evolution the observations concerning the individual with the highest median performance has been selected.

In the remaining parts of this chapter, we focus on the analysis of six individuals from the evolutionary runs 5, 6, 7, 8, 9 and 10. Since for the evolution with index 9 the type 2 individual is superior to the type 1 individual with respect to the median performance exhibited in the second phase of post-evaluation, we prefer to select the type 2 individual for further analysis. In all other cases, the type 1 individual is selected.

Transport Performance Concerning Different Shapes of Prey The 2,500 tests to evaluate the performance of the best individuals of each evolutionary run during the post-evaluation are based on prey objects of different shapes and masses. For every test the prey is modeled as a cuboid or as a cylinder; each of these shapes is selected with equal probability. The size of the prey is chosen randomly and independent from its mass (for details see the beginning of Section 3.4).

Although the geometrical properties of the prey have no influence on the magnitude of the frictional forces that arise during transport, the group's performance can be affected significantly by the shape and size of the prey. For example, larger prey offer more potential space to be grasped directly than smaller prey. We have observed that the evolved individuals concerning the experiments described in Chapter 2 exhibit an acceptable level of transport performance only if the prey item is big enough; in this case each member of the group of 2-5 s-bots that is trying to push the prey is able

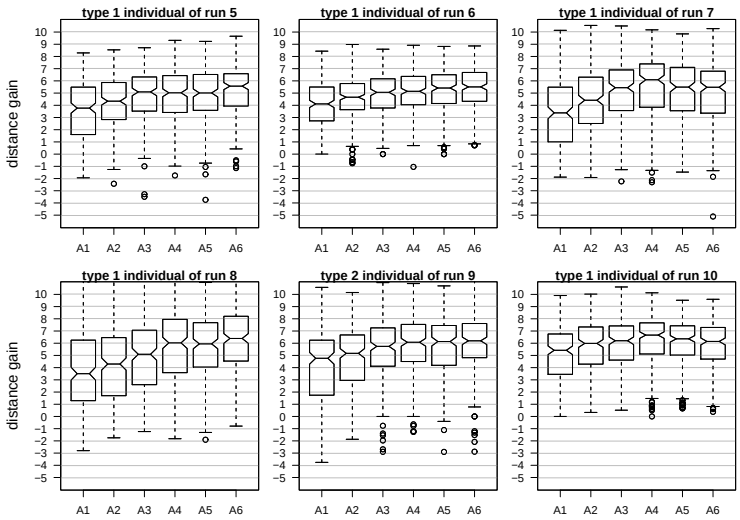

Figure 3.26.: Distance gain concerning transport of cylindrical prey objects for different intervals of radii referred by $A1, A2, \ldots, A6$ (the radii increases from the left to the right). All cylinders have a radius between 6 and 15 cm. For the six selected individuals of the evolutions 5, 6, 7, 8, 9 and 10, the distance gain concerning tests for the different sets is illustrated by a notched box-and-whisker plot. If the notches of two plots do not overlap then the medians are significantly different at the 5 percent level. Each box represents 205 to 229 observations.

to get in direct contact with the prey on the preferential[15] side (see Section 2.4.3).

To evaluate the robustness of the evolved controllers with respect to the shape and the size of the prey, the data concerning the 2,500 tests for the evaluation of the performance of the best individuals for each evolutionary run is partitioned according to different classes of prey:

- The sets $A1, A2, \ldots, A6$ comprise the data concerning cylindrical prey objects. Ai corresponds to tests in which the cylinders' radii are in range $[6 + 1.5(i - 1), 6 + 1.5i)$.

- The sets $B1, B2, \ldots, B6$ comprise the data concerning prey objects with cuboid shape. $B1, B2, \ldots, B6$ refer to tests in which the longer side length is within the ranges $[12, 16)$, $[16, 20)$, $[20, 25)$, $[25, 30)$, $[30, 35)$ and $[35, 40)$. The length of the shorter side is always in range of 12 to 20 cm.

- The sets H1 and H2 referring to the data concerning tests with prey of 12 cm and 20 cm height.

[15] The direction that is not heading to the beacon.

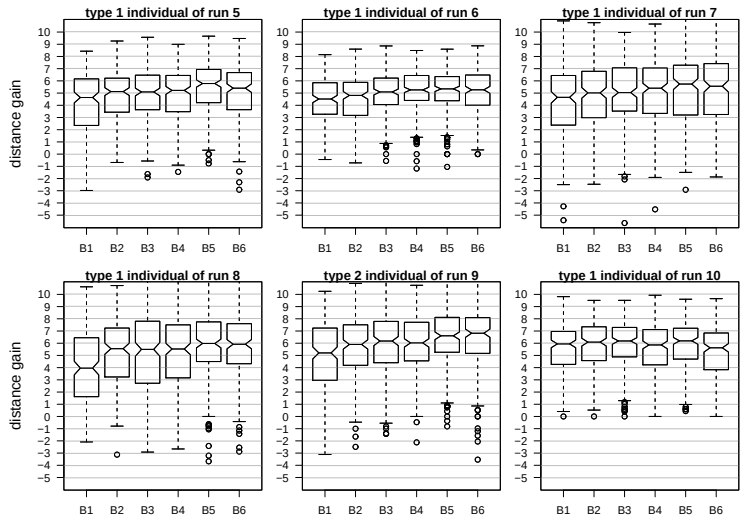

Figure 3.27.: Distance gain concerning transport of cuboid prey objects of different size: $B1, B2, \ldots, B6$ refer to six sets of tests in which cuboids have to be transported. The longer side of a each cuboid is 12 to 40 cm long. For the six selected individuals of the evolutions 5, 6, 7, 8, 9 and 10, the distance gain concerning tests for the different sets is illustrated by a notched box-and-whisker plot. Each box represents 188 to 215 observations.

Concerning the sets H1 and H2 the median performance observed for the individuals of the evolutions 5 to 10 are almost identical for the small prey objects (H1) and the tall ones (H2): the relative deviation is at most 2%. Due to the positioning method used, the s-bots are initially placed within a semi-circle around the prey. Since this semi-circle is oriented towards the beacon the s-bots can initially sense the target direction. The use of tall prey objects might lead to a performance decrease in case other positioning methods are used.

Figure 3.26 shows the observed distance gain concerning the sets $A1, A2, \ldots, A6$, i.e., for different ranges of radii of the cylindrical prey. It can be seen that most individuals perform quite robust with respect to the size of the cylinder. All individuals exhibit a performance decrease in case of the sets A1 and A2 corresponding to tests in which the cylinders' radii are less than 9 cm.

Figure 3.27 shows the observed distance gain concerning the sets $B1, B2, \ldots, B6$, i.e., for different length of the longer side of cuboid prey. It can be seen that most individuals perform fairly robust with respect to the length of the longer side. All but one individual exhibit a performance decrease in case of set B1 corresponding to tests in which the length of each cuboid's side is in the range of 12 to 16 cm.

Transport Performance Concerning a Different Initial Positioning As for the previous experiments we evaluate the robustness of the evolved controllers with respect to the full-circle positioning method that is utilized in the scalability analysis to place larger groups of s-bots in the vicinity of the prey (see Section 3.4.3.4). Changing the strategy of initial placement of s-bots near the prey has shown to diminish the group's performance of evolved controllers in previous experiments (see Section 3.4.3.1).

The selected individuals of the evolutionary runs 5, 6, 7, 8, 9 and 10 having the highest median performance are evaluated on 1,250 tests using the full-circle positioning method. The results are shown in Figure 3.28. The white boxes correspond to the original placement method. The (light and dark) gray boxes refer to the full-circle positioning.

Comparing the white and light gray boxes it can be seen that the new placement method results in a relatively high performance decrease for the individuals concerning the evolutions 5, 6, 8 and 9. The median distance gain of the individual of evolution 10 is slightly decreasing from 6.12 to 5.50. In case of the individual of evolution 7 the median distance even slightly increases from 5.31 to 5.87.

In both considered cases the tests last 20 seconds. If we extend the simulation period for 10 additional seconds, the median performances of all individuals increase again (dark gray boxes). Concerning the evolutions 5, 6 and 8, the additional time of 10 seconds is enough to compensate the loss of performance caused by the new placement strategy: the differences of the observed median distance gains concerning the white and dark gray boxes of the evolutions 5, 6 and 8 are rather low (-0.43, 0.04 and 0.1). In case of the other three evolutions the individuals perform even better, if the new placement strategy is used and 10 additional seconds are available: the median distance gains of the dark gray boxes for the evolutions 7, 9 and 10 exceed the median values of the corresponding white boxes by 2.12, 1.31 and 1.24. However, also the standard deviations increase. Concerning the observed distance gain values for the evolutions 5, 6, 7, 8, 9 and 10 the standard deviations using the full-circle positioning method and having 30 seconds of simulation time are 3.09 ($+0.81$), 2.67 ($+0.78$), 3.45 ($+0.65$), 3.43 ($+0.41$), 3.95 ($+1.29$) and 2.7 ($+0.5$). The numbers in the parentheses refer to the difference with respect to the semi-circle positioning strategy and a simulation time of 20 seconds.

In comparison with the performance attained in a similar[16] test by the individuals selected for the analysis concerning the evolutionary runs of the previous experiments (see Section 3.4.3.1) the controllers that have been generated in this experiment seem to be slightly more robust with respect to the new positioning strategy.

[16]In the previous experiments only prey objects of one specific shape and size are utilized.

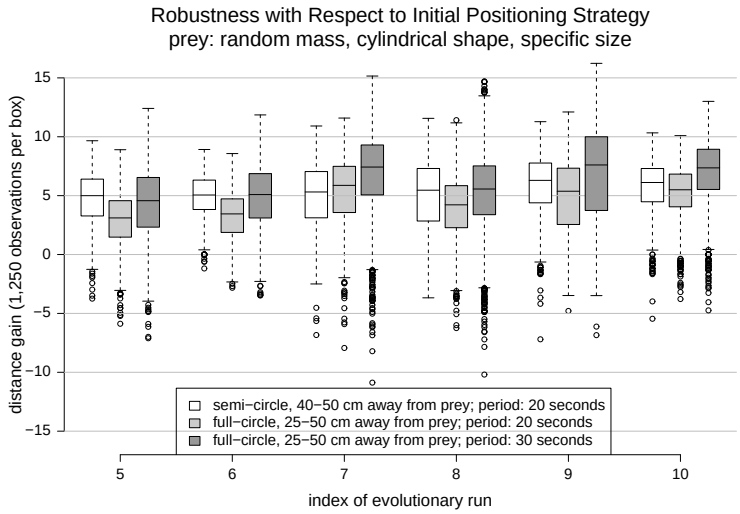

Figure 3.28.: Test to what extend the s-bot groups are robust with respect to two different initial placement methods. Each white box refers to 1,250 observations concerning the original initial placement that is used during the evolutionary runs. The other boxes refer to observations (1,250 each) for the *full-circle positioning* method.

3.4.3.2. Behavioral Analysis

The fitness function used in the evolutionary algorithm rewards the presence of assembled s-bot structures connected to the prey (see Section 3.3.3). To get a maximum reward, each s-bot should establish a connection either directly to the prey or to another s-bot that is directly or indirectly connected to the prey. Given a group of four s-bots there are 17 possible structural configurations of s-bots connected both to each other and to the prey.[17] These structures are illustrated in Figure 3.29.

For each configuration in the figure the frequency of occurrence of the particular structure at the end of the simulation and the corresponding median distance gain during the simulation concerning 2,500 tests of the type 1 individual of run 10 are displayed.

In Figure 3.30 and Figure 3.31 the frequencies and the distance gain values are plotted against the structures of the selected individuals of the evolutionary runs 9 and also for 10.

For both individuals all structures have appeared at least once; some with very low frequency. Comparing the two figures, it can be seen that structures that are composed of less than four s-bots (i.e., structures of indices 1-8) have appeared with higher frequency for the individual of run 9 (52.56%) than for the one of run 10

[17] Structures that are isomorphic count only once.

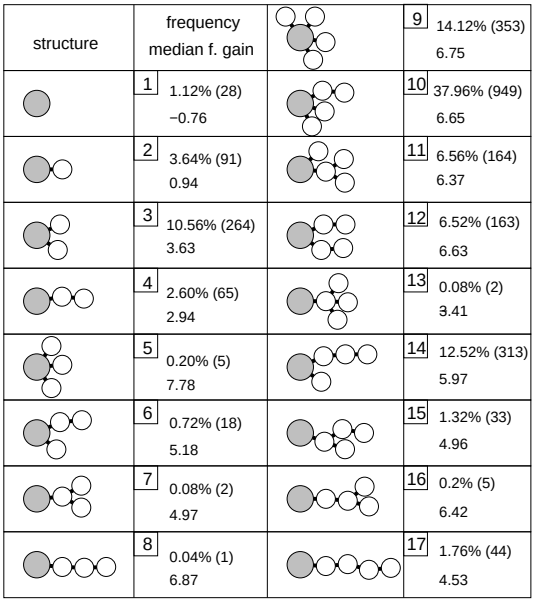

Figure 3.29.: List of possible structural configurations of four s-bots (directly or indirectly) connected both to each other and the prey. Concerning 2,500 tests for the type 1 individual of evolutionary run 10, the frequency of and the median distance for every particular structure is presented. Note that the observed structures are recorded at the end of the simulation, while the distance gain is the result of the group behavior during the entire simulation.

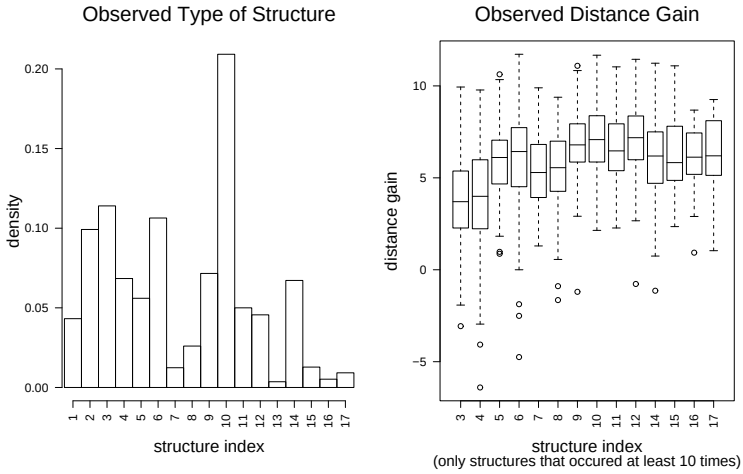

Figure 3.30.: Frequency and distance gain of structures observed during 2,500 tests with the individual of highest median performance for the evolutionary run 9. The structure indices correspond to the structures shown in Figure 3.29.

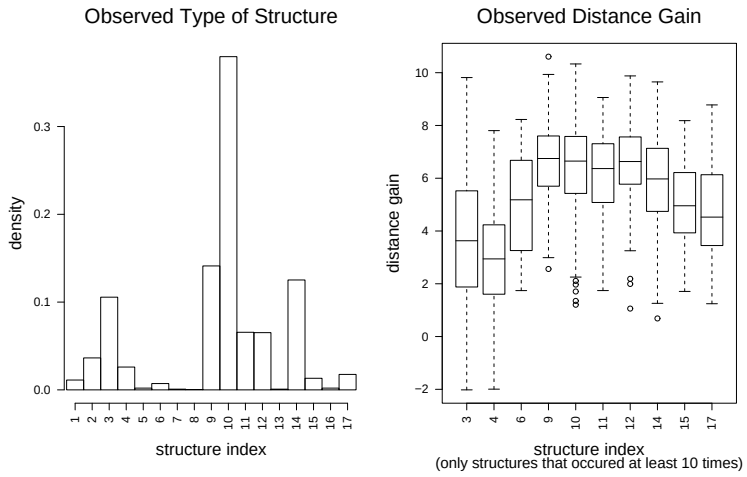

Figure 3.31.: Frequency and distance gain of structures observed during 2,500 tests with the individual of highest median performance of evolutionary run 10. The structure indices correspond to the structures shown in Figure 3.29.

(18.96%).

For almost all cases of structures including 3 or 4 s-bots (i.e., structures of indices from 5 to 17), at least two s-bots are directly connected to the prey (corresponding to the structure indices 5, 6, 9, 10, 11, 12 and 14). The cases in which only one single s-bot of an assembling structure is directly connected to the prey (i.e., structures of indices 7, 8, 13, 15, 16 and 17) appear rather seldom for both the individuals analyzed in this section and the individuals concerning the previous set of experiments (see Section 3.4.2.2).

On the other hand, structure 9, i.e., the configuration in which all four s-bots are connected directly to the prey has never been observed to be the most frequent one. In most cases just two or three connections are established; structure 10 appears the most often for all individuals that have been analyzed. Therefore, the individuals exploit both, multiple connections to the prey object as well as connections to other teammates.

3.4.3.3. Flexibility

In the experiments described in this chapter, the simulated environment is equipped with a single, light emitting beacon indicating the desired direction of transport. The beacon is placed 300 cm away from the prey. The neural network controllers synthesized within the evolutions let the group of s-bots form a structure in order to transport the prey into the desired direction.

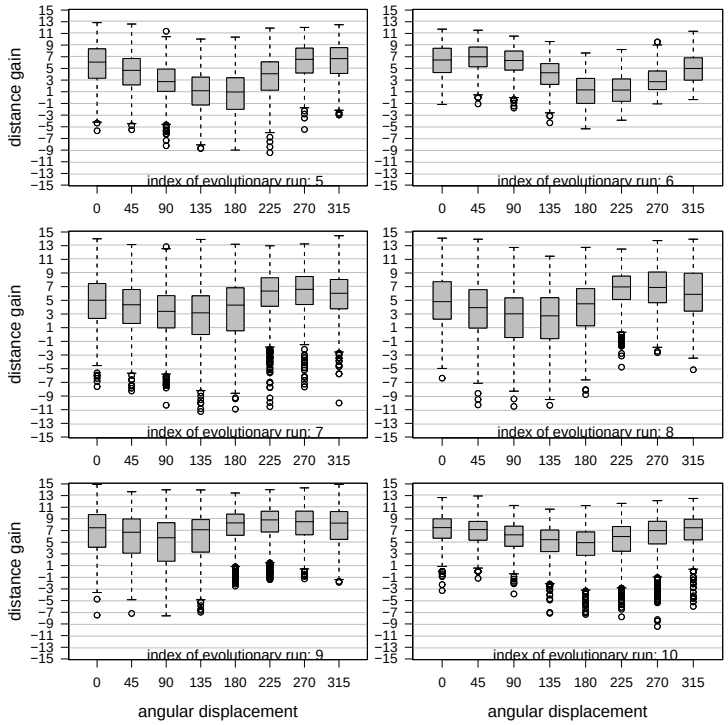

Figure 3.32.: Flexibility test: ability of a group engaged in the transport of a prey to adapt the direction of transport according to a new target. The distance gained within the second part of the simulation and with respect to the new target is illustrated (scaled by root functions). Eight different angular displacements between the old and the new target are considered.

Analog to an analysis concerning the previous experiments (see in Section 3.4.2.3) we evaluate to what extend a group of s-bots, engaged in the transport of a prey, is able to adapt dynamically the direction of transport according to a new target.

The simulation period of each test is doubled from 20 to 40 seconds. Concerning the first half of the simulation period, the basic experimental setup has been retained unchanged: the s-bots are placed within a semi-circle around the prey and the beacon is located 300 cm away from the prey. As soon as the first half of the simulation period is elapsed, the beacon (i.e., the target) is placed to a new position. The (horizontal) angular displacement of the beacon with respect to the prey can be 0, 45, 90, 135, 180, 225, 270 or 315 degrees (for details see Section 3.4.2.3).

For each angular displacement the selected individuals of the runs 5, 6, 7, 8, 9 and 10 have been evaluated on 1,250 tests.

As can be seen in Figure 3.32 the performance of all individuals is affected by

the angular displacements of the target. Similar to previous experiments (see Section 3.4.2.3) we observe a decrease of performance that is not symmetric concerning angular displacements to the left or to the right.

As long as the angle of displacement does not exceed 45% degrees to the left or to the right, the decreases in performance are quite low. The individuals concerning runs 9 and 10 obtained an acceptable level of performance for all angular displacements: the lowest median performances correspond to transport distances of 33 cm and 24 cm respectively.

3.4.3.4. Scalability

In this section we analyze the controllers' abilities to scale from group size 4 to the group sizes 8, 12 and 16. The mass of the prey is also modified linearly from 500 to 1,000, 1,500 and 2,000 g.

For each individual of the evolutionary runs 5-10 and for each group size 250 tests have been performed. The simulation period has been extended to 30 seconds (see also Section 3.4.2.4). To ensure a proper placement of groups up to 16 s-bots in the immediate vicinity of the prey, the full-circle positioning method (see Section 3.4.2.1) is utilized. Compared to the default placement strategy, this method has shown to cause a performance decrease for the majority of evolved individuals controlling a group of 4 s-bots to transport prey of masses from 500 to 1,000 g (see Section 3.4.3.1). However, we have observed that the selected individuals of the runs 5-10 are robust enough to reach an acceptable performance level, especially in case the simulation period is increased.

Figure 3.33 presents the results concerning 250 tests per individual and for each group size. Overall, the observed scaling abilities are not satisfying. It seems to be infeasible for the evolved individuals to let the s-bots transport prey of arbitrary high masses by adapting the group size linearly with respect to the mass.

However, it can be seen that for the particular group sizes of 4, 8, 12 and 16 some of the evolved individuals attain an acceptable performance level: the selected individuals of run 9 and 10 are able to transport all the prey objects with group sizes from 4 to 16 with an acceptable performance: in the majority of the observations concerning the group size 16 the prey has been moved for more than 43 cm and 31 cm respectively. Also the performances of the selected individuals of the run 5-8 are fairly good concerning group size 8 and is still acceptable for group up to 12 s-bots.

We suppose that the performance decrease for larger group sizes is partly caused by the increased fraction of s-bots that do not connect directly or indirectly to the prey. For the selected individuals of run 9 and 10 Figure 3.34 and Figure 3.35 display distributions of the number of s-bots that are part of the (connected) transport structure for different group sizes at the end of the simulation of each test. The number of observations in which more than 75% of the s-bots are part of the connected

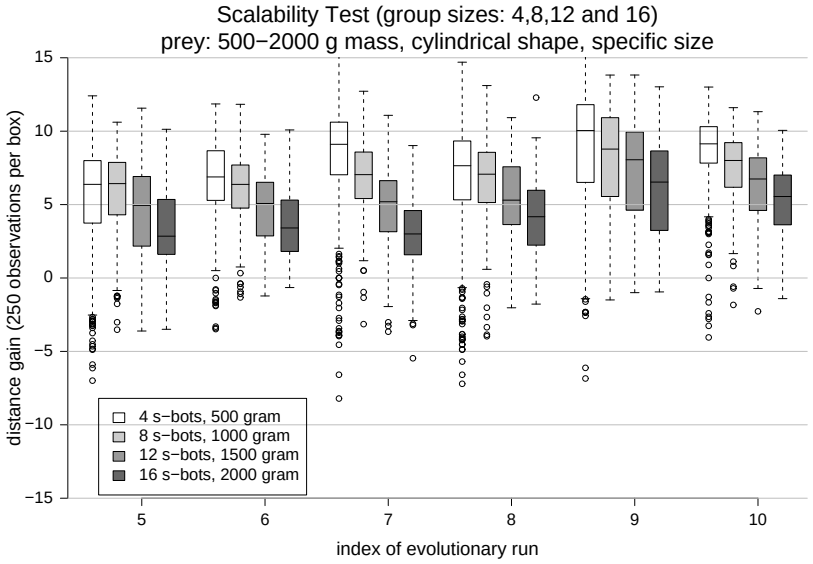

Figure 3.33.: Scalability test: distance gain within 30 seconds of simulation time with groups of size 4, 8, 12 and 16 transporting prey objects of mass 500, 1,000, 1,500 and 2,000 g.

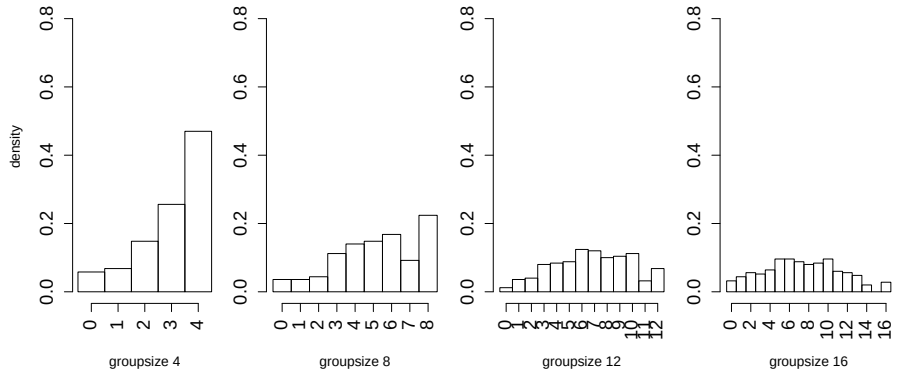

Figure 3.34.: Frequency of occurrence of different structure sizes: for the 1,250 tests in which the type 1 individual of run 9 controls a group of 4, 8, 12 or 16 s-bots in order to transport a prey of mass 500 to 2,000 g, the structure size, i.e., the number of s-bot directly or indirectly connected to the prey object has been recorded. Here, the frequencies that have been observed for each group size are given.

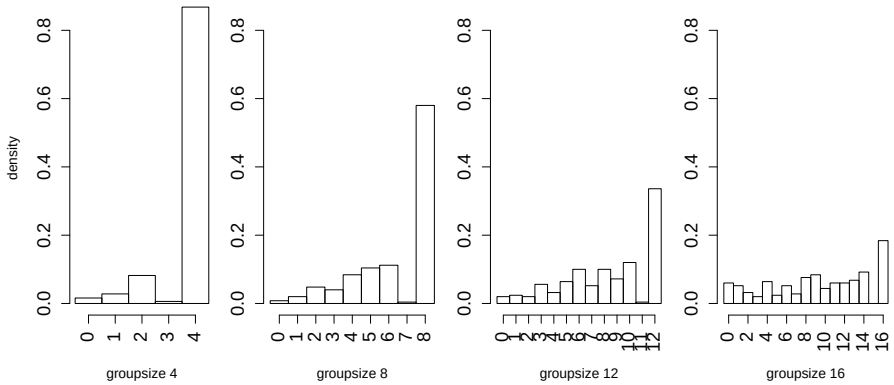

Figure 3.35.: Frequency of occurrence of different structure sizes: for the 1,250 tests in which the type 1 individual of run 10 controls a group of 4, 8, 12 or 16 s-bots in order to transport a prey of mass 500 to 2,000 g, the structure size, i.e., the number of s-bot directly or indirectly connected to the prey object has been recorded. Here, the frequencies that have been observed for each group size are given.

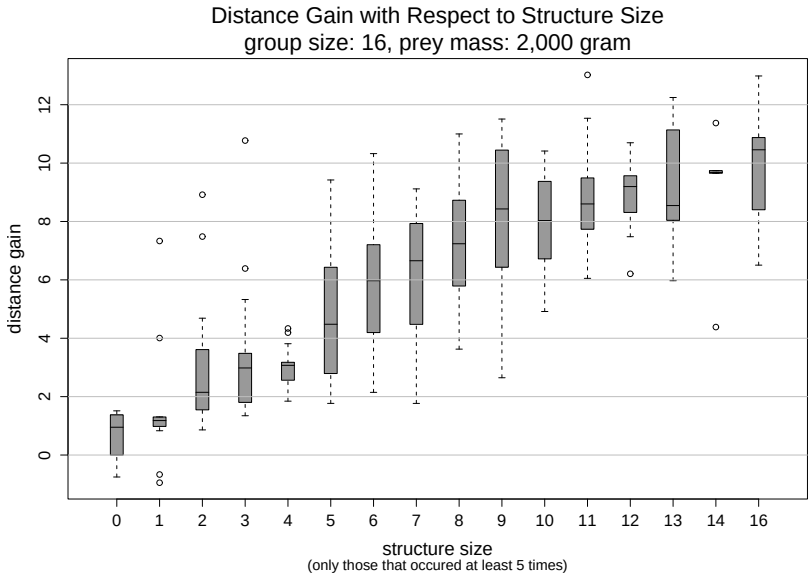

Figure 3.36.: Distance gain for different structure sizes. For the 750 tests of the selected individual of run 9 the distance gain is plotted against the size of the corresponding assembled transport structure observed at the final time step of the each test.

structure decreases continuously. The *skewness* observed for the distribution of group size 4 tends to get smaller for increased group sizes.

As we can see in Figure 3.36, structures of increased size perform better than smaller ones. In case of the tested combinations of group size and mass the prey can be retrieved even if some s-bots fail to get involved, since there is a redundancy. However, benefit could be gained if the fraction of s-bots actively engaged in the transport would be increased.

3.5. Conclusions

In this chapter we addressed the problem to control a group composed of 4 simple s-bots to cooperate to transport a heavy prey within a fixed time period as close as possible to a light emitting beacon. Depending on the mass that is varying among different prey objects, the prey can be moved efficiently only by cooperative behaviors of two, three or all four s-bots. Within two independent series of experiments both small prey of particular shape and size, and prey of different shapes and sizes are considered.

To control each s-bot of a group, a neural network controller synthesized by an artificial evolution is utilized. About the half of the evolutions have evolved fairly good performing individuals. These have been further analyzed.

For most solutions the transport performances (i.e., the distance gain with respect to the target) observed in 2,500 tests are fluctuating considerably. To some degree this is caused by the variation of mass among prey items ranging from 500 to 1,000 g. One evolved individual of the first series of experiment (i.e., the experiment with a particular shape and size of prey) exhibits remarkable few performance fluctuations and attains a respectable level of performance. The individuals of the second series of experiments act fairly robust with respect to different shapes and sizes of prey. Only in case of smaller prey objects, the performance of some controllers decreases considerably.

In order to move the prey, the s-bots form various assembled transport structures using the gripper element. The formation of structures has been emphasized by the fitness functions that have been used for the evolutions in both series of experiments. Concerning most observations, several direct connections between the prey and solitary or assembled s-bots have been established. In the majority of the observations, an assembled group of s-bots is part of the transport structure that is connected to the prey.

Most of the analyzed controllers fail to perform robust with respect to the full-circle positioning method (a strategy applied during the scalability analysis). Nevertheless, they are able to reach an acceptable level of performance if the simulation period is extended for 10 additional seconds.

Several controllers are able to respond efficiently to a dynamic repositioning of the target during simulation concerning eight different angular displacements. Some controllers fail to adapt the movement of the prey in case the new target is located in the opposite direction as the previous one (with respect to the prey).

The performance of all controller individuals decreases to some extend in case both group size and the mass of the prey are increased linearly by the factors 2, 3 or 4 (corresponding to groups composed of 4 to 16 s-bots transporting prey objects of mass 2,000 g). Nevertheless, some solutions attain still a level of performance that is acceptable for group sizes up to 16. However, we expect that these solutions cannot be applied successfully to groups much larger than 20 s-bots.

For the scalability analysis, the individuals have been exposed to tests in which a new positioning strategy (the full-circle positioning) is utilized to ensure a proper placement of larger groups of s-bots. This full-circle positioning has caused a decrease in performance for most individuals. Furthermore, we have observed that for bigger group sizes many s-bots are not actively engaged in the transport task. We expect that the efficiency, robustness and scalability of the evolved solutions could be improved by an incremental evolution. Here, a bigger group size of 8 to 10 s-bots and the full-circle positioning could be applied.

3.6. Future Work

In this work, we have designed neural networks controlling groups of simple s-bots engaged in the cooperative transport of heavy prey by using Artificial Evolution. Prey objects of different masses, shapes and sizes have been considered. The communication structure among s-bots have been limited to interactions via the environment or (additionally) interactions via sensing.

We have analyzed the obtained transport strategies with respect to desired properties such as efficiency, robustness, flexibility and scalability.

Some particular controllers are able to perform flexible and quite efficient for groups composed of size 2 to 16 s-bots. However, the performance of all solutions decreases with group size. It seems to be not feasible for any of the evolved individuals to let the s-bots transport prey of arbitrary high masses but constant size by adapting the group size linearly with respect to the mass. As discussed in the previous section, we expect that the efficiency, robustness and scalability of the evolved solutions could be improved by incremental evolution.

So far, we have considered the case that a group of predetermined size is placed in the immediate vicinity of a prey that has to be moved towards a target. In future we want to address the problem of cooperative prey retrieval in environments in which multiple prey items are present. Search and recruitment techniques will be of special interest. We want to design strategies that enable a swarm of s-bots

- to choose collectively the prey objects to be retrieved and
- to seek for an appropriate group size of s-bots engaged in the transport of each particular prey.

Last but not least, we want to control a group of real s-bots. Therefore either controllers will be transfered from the simulated to the real s-bots, or the evolutionary approach will be applied directly on the real s-bots. In any case, the real s-bot system will be used for further validation and improvements of the simulation model.

Appendix A.

Moment of Inertia

The moment of inertia I_a of a solid body of density $p(x)$ with respect to a given axis a is defined by the volume integral

$$I_a = \int p(x) r_a(x)^2 dx,$$

where $r_a(x)$ is the perpendicular distance of point x from the axis a of rotation.

The moment of inertia of a cuboid of mass m, side length a, b and c, and uniform density (see Figure A.1 (1)) for rotation about the three basic axis through the center of mass is given by:

$$\begin{aligned} J_x &= \frac{1}{12} m(b^2 + c^2), \\ J_y &= \frac{1}{12} m(a^2 + c^2), \\ J_z &= \frac{1}{12} m(a^2 + b^2). \end{aligned}$$

The moment of inertia for a cylinder of mass m, radius r, height h and uniform density (see Figure A.1 (2)) is given by:

$$\begin{aligned} J_x &= \frac{1}{12} mh^2 + \frac{1}{4} mr^2, \\ J_y &= \frac{1}{12} mh^2 + \frac{1}{4} mr^2, \\ J_z &= \frac{1}{2} mr^2. \end{aligned}$$

Lets consider an object composed of 2 rigidly connected cuboids B_1 and B_2 having a uniform density each (see Figure A.2 (1)). Both cuboids have identical side length a, b and c. Cuboid B_1 has a mass of m_1 and cuboid B_2 has a mass of b_2. Therefore the objects total mass is $m_1 + m_2$. Given a coordinate system with the point of origin

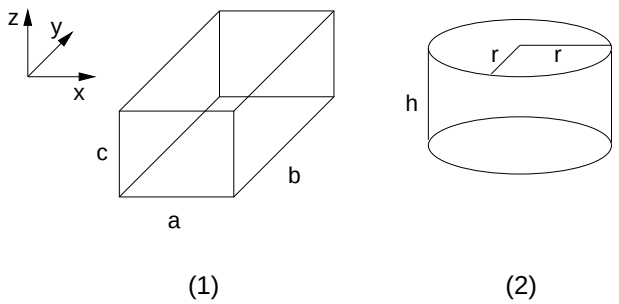

Figure A.1.: Prey objects with uniform density: (1) cuboid and (2) cylinder

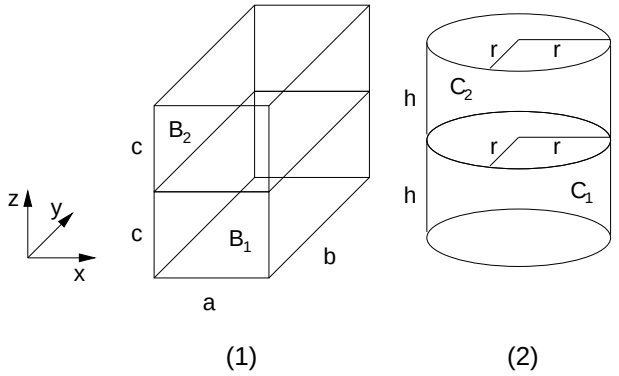

Figure A.2.: Objects composed of a lower part with mass m_1 and an upper part with mass m_2. Each part has a uniform density. (1) cuboid and (2) cylinder.

in the geometrical center of the object and the axis x, y and z oriented parallel to the sides a, b and c of the cuboids, the barycentre of the object is at $(0, 0, \frac{c}{2}\frac{m_2-m_1}{m_1+m_2})$.

If the moment of inertia J_a of a body about an axis a passing through its center of mass is known, the moment of inertia $J_{a'}$ of the body about another axis a' that is parallel to axis a can be determined as

$$J_{a'} = J_a + md^2,$$

where m is the mass of the body, and d is the separation of axes a and axes a'. This formula is known as the parallel-axis or Steiners theorem.

Applying Steiners theorem, the moment of inertia of the object shown in Figure A.2 (1) for rotation about the three basic axis through the barycentre of the object can be derived as:

$$J_x = \frac{1}{12}(m_1 + m_2)(b^2 + c^2) + m_1 \left(\frac{cm_2}{m_1 + m_2} \right)^2 + m_2 \left(\frac{cm_1}{m_1 + m_2} \right)^2$$
$$J_y = \frac{1}{12}(m_1 + m_2)(a^2 + c^2) + m_1 \left(\frac{cm_2}{m_1 + m_2} \right)^2 + m_2 \left(\frac{cm_1}{m_1 + m_2} \right)^2$$
$$J_z = \frac{1}{12}(m_1 + m_2)(a^2 + b^2)$$

The moment of inertia for the object composed of two cylinders of mass m_1 and m_2 and with radius r and height h each (see Figure A.2 (2)) for rotation about the three basic axis through the barycentre of the object is given by:

$$J_x = (m_1 + m_2)(\frac{1}{12}h^2 + \frac{1}{4}r^2) + m_1 \left(\frac{hm_2}{m_1 + m_2} \right)^2 + m_2 \left(\frac{hm_1}{m_1 + m_2} \right)^2$$
$$J_y = (m_1 + m_2)(\frac{1}{12}h^2 + \frac{1}{4}r^2) + m_1 \left(\frac{hm_2}{m_1 + m_2} \right)^2 + m_2 \left(\frac{hm_1}{m_1 + m_2} \right)^2$$
$$J_z = \frac{1}{2}(m_1 + m_2)r^2$$

Bibliography

Anderson, C. (2002). Self-organization in relation to several similar concepts: Are the boundaries to self-organization indistinct? *The Biological Bulletin 202*(3), 247–255.

Arkin, R. (1992). Cooperation without communication: Multiagent schema-based robot navigation. *Journal of Robotic Systems 9*(3), 351–364.

Asama, H., A. Matsumoto, and Y. Ishida (1989). Design of an autonomous and distributed robot system: Actress. In *Proceedings of IEEERSJ International Workshop on Intelligent Robots and Systems*, Tsukuba, Japan, pp. 283–290.

Ashby, W. (1947). Principles of the self-organizing dynamic system. *Journal of General Psychology 37*, 125.

Balch, T. (1997a). Behavioral diversity in robot teams. In *In Collected Papers from the 1997 AAAI Workshop on Multiagent Learning*, pp. 7–12. Cambridge, MA: AAAI Press.

Balch, T. (1997b). Social entropy: a new metric for learning multi-robot teams. In *Proceedings of 10th International Florida Artificial Intelligence Reseach Symposium*, Daytona Beach, FL, pp. 272–277. Florida AI Research Society.

Balch, T. (1999). The impact of diversity on performance in multi-robot foraging. In *Proceedings of the Third International Conference on Autonomous Agents (Agents'99)*, Seattle, WA, pp. 92–99. ACM Press.

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

Banzhaf, W. (2002). Self-organizing systems. In *Encyclopedia of Physical Science and Technology*, Volume 14, pp. 589–598. New York, NY: Academic Press.

Banzhaf, W., P. Nordin, R. Keller, and F. Francone (1998). Genetic programming — an introduction.

Becker, R., O. Holland, and J.-L. Deneubourg (1994). From local actions to global tasks: Stigmergy and collective robotics. In *Proc. of the fourth International Workshop on the Synthesis and Simulation of Living Systems*. MIT Press.

Beni, G. (1988). The concept of cellular robotic system. In *Proceedings of Third IEEE International Symposium on Intelligent Control*, Arlington, VA, pp. 57–62.

Beyer, H.-G. (1993). Toward a theory of evolution strategies: Some asymptotical results from the $(1, +\lambda)$-theory. *Evolutionary Computation 1*(2), 165–188.

Beyer, H.-G. (1994). Towards a theory of evolution strategies: Progress rates and quality gain for $(1, +\lambda)$ -strategies on (nearly) arbitrary fitness functions. In *Parallel Problem Solving from Nature*, Volume 3, pp. 58–67.

Beyer, H.-G. (1995). Toward a theory of evolution strategies: On the benefit of sex - the $(\mu/\mu, lambda)$-theory. *Evolutionary Computation 3*(1), 81–111.

Beyer, H.-G. (1996). Toward a Theory of Evolution Strategies: Self-Adaptation. *Evolutionary Computation 3*(3), 311–347.

Billard, A., A. Ijspeert, and A. Martinoli (1999). A multi-robot system for adaptive exploration of a fast changing environment: Probabilistic modelling and experimental study. *Connection Science 11*, 359–379.

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

Bond, A. H. and L. Gasser (1988). *Readings in Distributed Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann Publishers.

Camazine, S., J.-L. Deneubourg, N. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau (2001). *Self-Organization in Biological Systems*. Princeton: Princeton University Press.

Cao, Y. U., A. S. Fukunaga, and A. B. Kahng (1997, March). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots 4*(1), 7–23.

Chance, M. R. and C. J. Jolly (1970). *Social groups of monkeys, apes and men*. London: Dutton.

Coveney, P. and R. Highfield (1995). *Frontiers of Complexity - The Search for Order in a Chaotic World*. Ballantine Books, Inc.

Cramer, R. (1985). A representation for the adapative generation of simple sequential programs. In *Proceedings of an International Conference on Genetic Algorithms and the Applications*, pp. 183–187.

Şahin, E., T. Labella, V. Trianni, J.-L. Deneubourg, P. Rasse, D. Floreano, L. Gambardella, F. Mondada, S. Nolfi, and M. Dorigo (2002, October 6-9,). SWARM-BOTS: Pattern formation in a swarm of self-assembling mobile robots. In A. El Kamel, K. Mellouli, and P. Borne (Eds.), *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Hammamet, Tunisia. Piscataway, NJ: IEEE Press.

Darwin, C. (1859). *On the Origin of Species by Means of Natural Selection.* London.

Deneubourg, J. and S. Goss (1989). Collective patterns and decision-making. *Ethology, Ecology and Evolution 1*, 295–311.

Deneubourg, J., S. Goss, G. Sandini, F. Ferrari, and P. Dario (1990). Self-organizing collection and transport of objects in unpredictable environments. In *Proceedings of Japan - U.S.A Symposium on Flexible Automation*, Kyoto, Japan, pp. 1093–1098.

Deneubourg, J.-L., S. Goss, N. Franks, and J. Pasteels (1989). The blind leading the blind: modeling chemically mediated army ant raid patterns. *Journal of Insect Behaviour 2*, 719–725.

Dobzhansky, T. (1937). *Genetics and the Origin of Species.* Columbia University Press.

Donald, B. R., J. Jennings, and D. Rus (1994). Analyzing teams of cooperating mobile robots. In *Proceedings - IEEE International Conference on Robotics and Automation*, Volume 3, pp. 130–135. IEEE.

Drogoul, A. and J. Ferber (1993). From tom thumb to the dockers: Some experiments with foraging robots. In *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, Cambridge, MA, pp. 451–459. MIT Press/Bradford Books.

Durfee, E. H. and J. S. Rosenschein (1994). Distributed problem solving and multiagent systems: Comparisons and examples. In M. Klein (Ed.), *Proceedings of the 13th International Workshop on DAI*, Lake Quinalt, WA, USA, pp. 94–104.

Elman, J. (1990). Finding structure in time. *Cognitive Science 14*, 179–211.

Fogel, L. (1962). Toward inductive inference automata. In *Proceedings of the Int. Federation for Information Processing Congress Conference*, pp. 395–399.

Fogel, L., A. Owens, and M. Walsh (1966). *Artificial Intelligence through Simulated Evolution.* John Wiley & Sons.

Fukada, T. and S. Nakagawa (1987). A dynamically reconfigurable robotic system. In *Proceedings of the International Conference on Industrial Electronics, Control, and Instrumentation (IECON)*, pp. 588–595.

Gasser, L. and M. N. Huhn (1989). *Distributed Artificial Intelligence*, Volume 2 of *Research Notes in Artificial Intelligence.* London, UK: Pitman Publishing and Morgan Kaufmann Publishers.

Goldberg, D. and M. J. Matarić (1999). Coordinating mobile robot group behavior using a model of interaction dynamics. In O. Etzioni, J. P. Müller, and J. M. Bradshaw (Eds.), *Proceedings of the Third International Conference on Autonomous Agents (Agents'99)*, Seattle, WA, USA, pp. 100–107. ACM Press.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley Publishing Company, Inc.

Goss, S. and J.-L. Deneubourg (1992). Harvesting by a group of robots. In *Proceedings of the First European Conference on Artificial Life*, Cambridge, MA, pp. 195–204. MIT Press/Bradford Books.

Grassé, P. (1959). La reconstruction du nid et les coordinations inter-individuelles chez bellicoitermes natalenis et cubitermes sp. la theorie de la stigmergie: Essai d'interpretation des termites constructeurs. *Insect Societies 6*, 44–83.

H.-G. Beyer (2001). *The Theory of Evolution Strategies*. Springer Verlag.

Haken, H. (1988). *Information and Self-organization. A Macroscopic Approach to Complex Systems*. Springer-Verlag.

Hayes, A., A. Martinoli, and R. Goodman (2000). Comparing distributed exploration strategies with simulated and real autonomous robots. In *Proceedings of the Fifth International Symposium on Distributed Autonomous Robotic Systems DARS-00*, Berlin, Germany, pp. 261–270. Springer-Verlag.

Heylighen, F. (2002). The science of self-organization and adaptivity. In *The Encyclopedia of Life Support Systems*. EOLSS Publishers Co. Ltd.

Holland, J. H. (1962). Outline for a logical theory in adaptive systems. *Journal of the Association for Computing Machinery 9*(3), 297–314.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. MIT Press.

Hölldobler, B. (1983). Territorial behavior in the green tree ant (oecophylla smaragdina). *Biotropica 15*, 241–250.

Hölldobler, B. and E. Wilson (1990). *The Ants*. Cambridge: Harvard University Press.

Huhns, M. N. (1987). *Distributed Artificial Intelligence*, Volume 1 of *Research Notes in Artificial Intelligence*. London, UK: Pitman Publishing and Morgan Kaufmann Publishers.

Kato, T. and D. Floreano (2001). An evolutionary active-vision system. In *Proceedings of the Congress on Evolutionary Computation*, Piscataway. IEEE-Press.

Koza, J. R. (1992). *Genetic Programming*. MIT Press.

Kube, C. and H. Zhang (1995). Stagnation recovery behaviours for collective robotics. In *Proceedings 1994 IEEE/RSJ/GI International Conference on Intelligent Robotics and Systems*, Los Alamitos, CA, pp. 1883–1890. IEEE Computer Society Press.

Kube, C. R. and H. Zhang (1992a). Collective robotic intelligence. In *Second International Conference on Simulation of Adaptive Behaviour*, pp. 460–468.

Kube, R. C. and E. Bonabeau (2000). Cooperative transport by ants and robots. *Robotics and Autonomous Systems 30*(Issue 1/2), 85–101. ISSN: 0921-8890.

Kube, R. C. and H. Zhang (1992b). Collective robotics: from social insects to robots. *Adaptive Behaviour, 1994 2*(2), 189–218.

Kuniyoshi, Y., N. Kita, S. Rougeaux, S. Sakane, M. Ishii, and M. Kakikura (1994). Cooperation by observation - the framework and basic task patterns. In *Proceedings 1994 IEEE Int. Conf. on Robotics and Automation*, San Diego, California, pp. 767–774.

Lamarck, J. B. (1809). *Philosophie zoologique*. Paris.

Markon, S., D. V. Arnold, T. Baeck, T. Beielstein, and H.-G. Beyer (2001, 27-30). Thresholding - a selection operator for noisy ES. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, pp. 465–472. IEEE Press.

Marocco, D. and D. Floreano (2002). Active vision and feature selection in evolutionary behavioral systems. In J. Hallam, D. Floreano, G. Hayes, and J. Meyer (Eds.), *From Animals to Animats 7: Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior*, Cambridge, MA, pp. 247–255. MIT Press-Bradford Books.

Martinoli, A. and K. Easton (2003). Optimization of swarm robotic systems via macroscopic models. In *Proceedings of the Second International Workshop on Multi-Robots Systems*, Washington, DC, pp. 181–192. Kluwer Academic Publishers.

Martinoli, A. and F. Mondada (1995). Collective and cooperative group behaviours: Biologically inspired experiments in robotics. In *Proceedings of the Fourth Symposium on Experimental Robotics ISER-95*, Stanford, USA.

Martinoli, A., M. Yamamoto, , and F. Mondada (1997). On the modelling of bioinspired collective experiments with real robots. In *Proceedings of the Fourth European Conference on Artificial Life ECAL-97*, Brighton, UK.

Mataric, M., M. Nilsson, and K. Simsarian (1995). Cooperative multi-robot box-pushing. In *Proceedings 1994 IEEE/RSJ/GI International Conference on Intelligent Robotics and Systems*, Los Alamitos, CA, pp. 556–561. IEEE Computer Society Press.

Mataric, M. J. (1992a). Designing emergent behaviors: From local interactions to collective intelligence. In *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, Honolulu, Hawaii, pp. 432–441. MIT Press.

Mataric, M. J. (1992b). Minimizing complexity in controlling a mobile robot population. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice, France, pp. 830–835.

Mataric, M. J. (1994, May). *Interaction and Intelligent Behavior*. Ph. D. thesis, MIT.

Mataric, M. J. (1995, December). Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems 16*, 321–331.

Mayr, E. (1942). *Systematics and the origin of species*. Columbia University Press.

Mayr, E. (1984). Evolution. *Spektrum der Wissenschaft*, 8–19.

Mendel, G. J. (1866). Versuche über Pflanzen-Hybriden. In *Verhadlungen des Naturforschenden Vereines in Brünn*, Volume 4, Czechoslovakia, pp. 3–47. Natural History Society of Brünn. Engl. title: Experiments in plant hybridization.

Moffet, M. (1988). Cooperative food transport by an asiatic ant. *National Geog. Res. 4*, 386–394.

Mondada, F., A. Guignard, A. Colot, D. Floreano, J.-L. Deneubourg, L. Gambardella, S. Nolfi, and M. Dorigo (2002, March). Swarm-bot: A new concept of robust all-terrain mobile robotic system. Technical report, LSA2 - I2S - STI, Swiss Federal Institute of Technology, Lausanne, Switzerland.

Mondada, F., G. C. Pettinaro, I. Kwee, A. Guignard, L. Gambardella, D. Floreano, S. Nolfi, J.-L. Deneubourg, and M. Dorigo (2002, September 8-13,). SWARM-BOT: A swarm of autonomous mobile robots with self-assembling capabilities. In C. Hemelrijk and E. Bonabeau (Eds.), *Proceedings of the International Workshop on Self-organisation and Evolution of Social Behaviour*, Monte Verità, Ascona, Switzerland, pp. 307–312. University of Zurich.

Mondada, R., E. Franzi, and P. Ienne (1993). Mobile robot miniaturization: A tool for investigation in control algorithms. In *Proceedings of the Third International Symposium on Experimental Robots*, Berlin. Springer-Verlag.

Nicolis, G. and I. Prigogine (1977). *Self-Organization In Non-Equilibrium Systems*. New York: Wiley.

Nolfi, S. (1998). Evolutionary robotics - exploiting the full power of self-organization. *Connection Science 10*(3-4), 167–184.

Nolfi, S. and D. Marocco (2002). Active perception: A sensorimotor account of object categorization. In *From animals to animats 7 - The Seventh International Conference on the Simulation of Adaptive Behavior*. The MIT Press.

Parker, L. (1994). *Heterogeneous Multi-Robot Cooperation*. Ph. D. thesis, MIT.

Parker, L. (1998). Alliance: An architecture for fault-tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation 14*(2), 220–240.

Parker, L. (1999). Adaptive heterogeneous multi-robot teams. *Neurocomputing 28*, 75–92. Special issue of NEURAP 98: Neural Networks and Their Applications.

Parker, L. E. (2000, October). Current state of the art in distributed autonomous mobile robotics. In L. E. Parker, G. Bekey, and J. Barhen (Eds.), *Distributed Autonomous Robotic System 4*, Tokyo, pp. 3–12. Springer-Verlag. ISBN: 4-431-70295-4.

Premvuti, S. and S. Yuta (1990). Consideration on the cooperation of multiple autonomous mobile robots. In *Proceedings of the IEEE International Workshop of Intelligent Robots and Systems*, Tsuchiura, Japan, pp. 59–63.

Rechenberg, I. (1965). *Cybernetic solution path of an experimental problem.* Ph. D. thesis, Royal Aircraft Establishment, Hants Farnborough, UK. Reprinted in 'Evolutionary Computation : The fossil record', D. B. Fogel, Ed., chap. 8, pp. 297-309, IEEE Press 1998.

Rechenberg, I. (1973). *Evolutionsstrategie. Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Frommann-Holzboog.

Robson, S. and J. Traniello (1998). Resource assessment, recruitment behavior, and organization of cooperative prey retrieval in the ant formica schaufussi (hymenoptera: Formicidae). *J. Insect Behavior 11*, 1–22.

Schwefel, H.-P. (1974). Adaptive Mechanismen in der biologischen Evolution und ihr Einfluß auf die Evolutionsgeschwindigkeit. Technical report, TU Berlin.

Schwefel, H.-P. (1975). *Evolutionsstrategie und numerische Optimierung.* Ph. D. thesis, TU Berlin.

Sen, A., M. Sekaran, and J. Hale (1994). Learning to coordinate without sharing information. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, Menlo Park, CA, pp. 426–431.

Shannon, C. and W. Weaver (1949). *The Mathematical Theory of Communication.* Urbana, Illinois: University of Illinois Press.

Sudd, J. (1963). How insects work in groups. *Discovery 24*, 15–19.

Sudd, J. (1965). The transport of prey by ant. *Behaviour 25*, 234–271.

Traniello, J. and S. Beshers (1991). Maximization of foraging efficiency and resource defense by group retrieval in the ant formica schaufussi. *Behav. Ecol. Sociobiol. 29*, 283–289.

Trianni, V., R. Groß, T. H. Labella, E. Şahin, P. Rasse, J.-L. Deneubourg, and M. Dorigo (2003). Evolving aggregation behaviors in a swarm of robots. In *ECAL 2003: Proceedings of the 7th European Conference on Artificial Life.* Springer-Verlag. to appear.

Tung, B. and L. Kleinrock (1993). Distributed control methods. In *Proceedings of the 2nd International Symposium on High Performance Distributed Computing*, Washington, pp. 206–215.

Acknowledgments

I want to express my thanks to all my colleagues at IRIDIA for always being kind, helpful and open for fruitful discussions. Special thanks to Shervin Nouyan, Vito Trianni, Keno Albrecht, Martin Villwock and Martin Fiedler to give many suggestions improving this work.