Available online at www.sciencedirect.com
SCIENCE DIRECT
Neural Networks 19 (2006) 705–720
Neural Networks
www.elsevier.com/locate/neunet

Modelling brain emergent behaviours through coevolution of neural agents

Michail Maniadakis[a,b], Panos Trahanias[a,b,*]

[a]Institute of Computer Science, Foundation for Research and Technology—Hellas (FORTH), P.O. Box 1385, Heraklion, 711 10 Crete, Greece
[b]Department of Computer Science, University of Crete, P.O. Box 1470, Heraklion, 714 09 Crete, Greece

Received 13 April 2004; accepted 25 February 2005

Abstract

Recently, many research efforts focus on modelling partial brain areas with the long-term goal to support cognitive abilities of artificial organisms. Existing models usually suffer from heterogeneity, which constitutes their integration very difficult. The present work introduces a computational framework to address brain modelling tasks, emphasizing on the integrative performance of substructures. Moreover, implemented models are embedded in a robotic platform to support its behavioural capabilities. We follow an agent-based approach in the design of substructures to support the autonomy of partial brain structures. Agents are formulated to allow the emergence of a desired behaviour after a certain amount of interaction with the environment. An appropriate collaborative coevolutionary algorithm, able to emphasize both the speciality of brain areas and their cooperative performance, is employed to support design specification of agent structures. The effectiveness of the proposed approach is illustrated through the implementation of computational models for motor cortex and hippocampus, which are successfully tested on a simulated mobile robot.

Keywords: Coevolution; Brain modelling; Robotics; Self-organized NNs; Hebbian learning; Phylogenesis; Epigenesis

1. Introduction

During the last decades, many research efforts have focused on the development of intelligent machines, which attempt to mimic the performance of biological organisms. Since mammals constitute the category of biological organisms that exhibit the highest level of intelligence, they could be used as an excellent prototype for the development of machines with advanced cognitive abilities. Modern theories for the explanation of mammalian cognition argue that the observed behaviour of animals is a result of phylogenetic development, and subjective environmental experience (Geary & Huffman, 2002). Evidently, this argument may also form a basis for the development of cognition in artificial organisms (Tempesti, Roggen, Sanchez, & Thoma, 2002).

To better understand the behaviour of animals, it is necessary to appreciate how their brain is functionally and anatomically organized. Cognitive capabilities of mammals are supported by their central nervous system (CNS). The latter consists of several interconnected modules with different functionalities (Kandel, Schwartz, & Jessell, 2000). A lot of research is recently oriented towards determining how the information flows within CNS modules, what kind of information is processed in each area of the CNS, and how these modules cooperate in order to accomplish real world tasks (Cotterill, 2001).

Even if the detailed, exact properties of each brain module in mammals are not clear yet, a large number of computational models of mammalian brain areas have been proposed capturing the known characteristics of these structures, as a means to explain and reproduce their functionality (Ajemian, Bullock, & Grossberg, 2000; Arleo & Gerstner, 2000; Kali & Dayan, 2000; Maniadakis & Trahanias, 2003; Norman & O'Reilly, 2001; Samsonovich & McNaughton, 1997; Stringer, Rolls, & Trappenberg, 2004; Todorov, 2000). These models operate at different

* Corresponding author. Address: Institute of Computer Science, Foundation for Research and Technology—Hellas (FORTH), P.O. Box 1385, Heraklion, 711 10 Crete, Greece. Tel.: +30 2810 391 715; fax: +30 2810 391 601.
E-mail addresses: mmaniada@ics.forth.gr (M. Maniadakis), trahania@ics.forth.gr (P. Trahanias).

0893-6080/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2005.02.007

levels of description and explanation, based on different assumptions. At the same time, very different computational approaches are followed by researchers in these modelling endeavours, which are usually constrained from heuristic design specifications. The developed models seem to form a heterogeneous collection, where computational differences among them constitute their integration very difficult (Wermter & Sun, 2000). Thus, a consistent procedure to support design specifications and model integration is still lacking.

In the present work, we introduce a systematic method to implement computational models of each part of the mammalian central nervous system, emphasizing on the integrative performance of substructures. Moreover, following research efforts which link cognitive capabilities of robots to brain science (Scassellati, 2000; Shin, 2002), implemented CNS models are embedded in a robotic platform to furnish it with behavioural capabilities. Thus, interaction with the environment is supported, and the proposed approach can be assessed. By employing a computational model, which follows the mammalian paradigm in both the functional organization and the mechanisms for cognition development, we aim at the construction of artificial systems able to develop advanced cognitive capabilities. It is worth emphasizing that the current work addresses primarily the development of a method to support CNS modelling for robotic applications, than presenting perfect models of specific brain areas.

We follow an agent-based approach to design models of partial brain areas, supporting the autonomy of substructures. This approach offers many advantages in terms of modular and scalable development of effective models (Franklin & Graesser, 1996; Jennings, 2000). Each agent consists of a neural network structure that captures the basic anatomical principles of the mammalian CNS. The design of agents focuses on the emergence of partial brain model functionality, based on robot–environment interaction (Cotterill, 2001; Thelen, 2000). Similar to a phylogenetic process, we employ an evolutionary approach to specify the computational details for each neural agent (Rolls & Stringer, 2000). Instead of using a unimodal evolutionary process, we employ a collaborative coevolutionary method which offers enhanced search abilities of partial elements, and emphasizes the independence of agent structures (Maniadakis & Trahanias, 2004). Moreover, the coevolutionary approach is also able to emphasize the cooperative performance of partial brain models (Poter & De Jong, 2000) and, therefore, the ability to achieve integration of partial models is inherent to our approach.

However, there are still open issues in the area of coevolutionary methods and current research efforts attempt to address them (Wiegand, Liles, & De Jong, 2001). One major problem concerns how collaborators are chosen among species, especially in the case of more than two coevolved species (Wiegand, Liles, & De Jong, 2002). We propose in this work a two level collaborative coevolutionary strategy (Maniadakis & Trahanias, 2004), aiming at a systematic method to approach this issue. Following our method, coevolution of a large number of species is possible. Thus, it can be used for large scale brain modelling tasks, with separate species for each partial brain structure.

The rest of the paper is organized as follows. In Section 2, we provide the motivation behind our work. Then Section 3 presents a formal computational framework for the implementation of partial brain models. In Section 4 we present a collaborative coevolutionary scheme with purposeful selection of collaborators, which is able to support coevolution of a large number of species. Computational experiments which follow the proposed approach to accomplish two partial brain modelling tasks are presented in Section 5. Both models are embodied in a robotic platform to furnish it with cognitive abilities and verify the validity of results. Specifically, we demonstrate the implementation of a computational model of motor cortex able to achieve a wall avoidance navigation behaviour, and also a computational model of hippocampus which achieves a self-localization behaviour by means of development of place cells. A detailed discussion on the achieved results ends this section. Finally, conclusions and suggestions for further work are drawn in the last section.

2. Motivation

The long-term vision of developing artificial organisms with high cognitive abilities, has given new impetus in brain modelling studies. In this endeavour, environmental interaction is of utmost importance, since it is difficult to investigate the mammalian CNS without embedding the models into a body to interact with its environment. Our work aims at supporting both intelligence development in artificial organisms and brain modelling efforts, by bringing them in the same field.

An agent-based approach seems suitable to support brain modelling tasks, mainly due to the distributed organization of CNS. Agents are deemed as a new theoretical tool for modelling complex, distributed systems. Agent-based technology is appropriately designed to facilitate the development of intelligent systems with a large number of cooperative interactive parts, supporting their flexibility, autonomy, subjectivity, and situatedness in a specific environment (Franklin & Graesser, 1996). From a designer point of view, it supports problem decomposition, abstraction of partial models, and scalability of global problem solution (Jennings, 2000).

The design of agent structures should ideally be based on the natural principles of the CNS of biological organisms. Recently, there is a debate among genetics and neurobiology regarding the extent that brain organization and the associated cognitive functions are genetically predetermined, or emerge through patterns of developmental

experience (for a recent review, see Geary and Huffman, 2002). One proposition claims that brain structure is the result of an evolutionary process over time (Duchaine, Cosnides, & Tooby, 2001). At the same time, recent research has provided evidence for the fact that the observed behaviour of mammals is a result of their continuous interaction with the environment throughout their lives (Cotterill, 2001; Thelen, 2000). In contrast to the widely used human-oriented hardwired solutions that support cognition of artificial systems, the emphasis on environmental experience highlights subjective understanding of the organism about the world.

Based on the above, it seems plausible that both genetically encoded features and subjective experience have a significant role in the formation of the animal's behaviour. Phylogenetic development determines the internal dynamics of brain structures that allow the epigenetic[1] emergence of valuable behaviours, after a certain amount of interaction with the environment. Besides the modulation of epigenetic learning by phylogenesis, the reverse interaction is also true. The well known since 1896 Baldwin effect, discusses the outcome of epigenetic learning on evolution, with the best able to learn organisms having larger numbers of offsprings (Smith, 1987).

In the present work, we propose a computational approach to brain modelling tasks, which combines the above aspects. A collection of agents is employed to represent CNS modules. Their design is based on the interactions of phylogenetic and epigenetic processes. Phylogenesis is represented by an evolutionary process, while epigenesis is represented by online adjustment of CNS agents. The objective that is followed in agent evolution is to furnish them with abilities to develop performance similar to the respective brain areas, after a certain amount of interaction with the environment. In other words, the evolutionary process genetically determines the internal dynamics of partial brain models, which in turn allow the emergence of a desired behaviour during lifetime performance. In this context, the utilization of evolutionary processes for brain modelling has also been proposed by others (Rolls & Stringer, 2000), although not adequately tested.

Furthermore, evolutionary psychologists proposed the coevolution of partial brain areas over time (Klein, Cosmides, Tooby, & Chance, 2002). The computational analogy of this proposition is that the evolutionary approach can be further extended to a coevolutionary one. Using a coevolutionary method, the computational details for each agent are confined to a small searchable domain, without constraining its functional capabilities. We propose, therefore, the employment of a coevolutionary approach for the design of brain models, because compared to a unimodal evolutionary process it exhibits the following advantages (Poter & De Jong, 2000):

- It is able to support both the individual and the cooperative characteristics of partial brain areas.
- It offers a systematic methodology to deal with the integration of different structures.
- It supplies a mechanism for developmental brain modelling by adding gradually new coevolved species to represent new brain areas.
- Similarly to all evolutionary strategies, it is a biologically plausible method.

Additionally, the brain modelling problem fits very well to collaborative coevolutionary approaches, because separate coevolved species can be used to perform design decisions for each model of partial brain area. Hence, by using a distinct fitness function to evolve each species, different roles can be assigned to partial structures. At the same time, the concurrent evolution of each species under a common evolutionary scheme enforces cooperation among brain modules. Therefore, we argue that coevolution facilitates complex brain-modelling tasks.

In summary, the current paper introduces a novel method to approach the implementation of mammalian CNS computational models, which are embedded in a robotic platform to furnish it with cognitive abilities. We introduce a computational framework for brain modelling tasks emphasizing on the integrative performance of substructures. It is based on the fact that CNS consists of distinct areas with different functionalities. Each brain area is modelled by a flexible agent structure, to emphasize on the special characteristics of the area. The internal dynamics of each agent are specified by a phylogenetic process. The latter aims at allowing the development of the desired behaviour in each agent structure, after a certain amount of environmental interaction. The cooperation of partial structures is achieved by employing a coevolutionary mechanism to support integrative performance of agents. Following a coevolutionary approach, both the individual and the cooperative characteristics of brain areas are highlighted.

In the following, we present in turn the computational details of the neural agent model, and the coevolutionary scheme used to perform design decisions of the model's structure.

3. Computational model

We have implemented two different neural network based agents, to supply general computational structures for brain modelling: (a) a computational cortical agent to represent brain areas, and (b) a link agent to support information flow across brain areas.

The proposed computational structures are not restrictive for the coevolutionary method, but rather serve as a guide on

[1] Epigenesis here, includes all learning processes during lifetime.

how coevolutionary approaches can be employed to support brain modelling tasks. Currently, the employed computational structures have been formulated as simple configurations that are suitable for the tasks at hand. Additional constraints can be integrated to increase their biological reliability, or even more, a different neural structure with emphasis on biological features can be used instead, to implement CNS models with enhanced biological reliability.

Neural network agents which serve the needs of the present study, are presented in detail in the next sections.

3.1. Cortical agent

Each cortical agent consists of a population of excitatory and inhibitory neurons. A rectangular plane with both sets of neurons, uniformly distributed, simulates the cortical area. Thus, an excitatory and inhibitory grid are defined on the cortical plane with each neuron occupying a predefined position (Fig. 1).

In order to achieve common spatial properties for neurons in the middle and neurons in the borders of the plane, we assume that opposite planar sides are met and the neurons near by can be connected. Four sets of synapses are defined depending on the nature of presynaptic and postsynaptic neurons (excitatory–excitatory, excitatory–inhibitory, inhibitory–excitatory, inhibitory–inhibitory). The connectivity of neurons follows the general rule of locality (Redish, Elga, & Touretzky, 1996). Synapse formation in cortical agents (Fig. 1) is based on a circular neighbourhood measure with the possibility of a different radius for each of the four synapse sets. In that way, bi-directionally neural pairs can be defined, with the flexibility of assuming different synaptic weights in each direction.

Both excitatory and inhibitory neuron sets follow the Wilson–Cowan model with sigmoid activation, similar to Tkaczyk (2001). The firing rate x of a neuron is updated based on the afferent input information A and the excitatory E and inhibitory I signals accepted by neighbouring neurons. This is expressed mathematically in a single form for both types of neurons, by

$$\mu\,\Delta x = -x + S(W_A A + W_E E - W_I I) \quad (1)$$

where μ presents the membrane time constant, W_A are the synaptic weights of the afferent signals, and W_E, W_I the synaptic weights of neighbouring excitatory and inhibitory neurons. $S(y) = 1/(1 + e^{-a(y-b)})$ is the non-linear sigmoid function where β and α stand for the threshold and the slope, respectively.

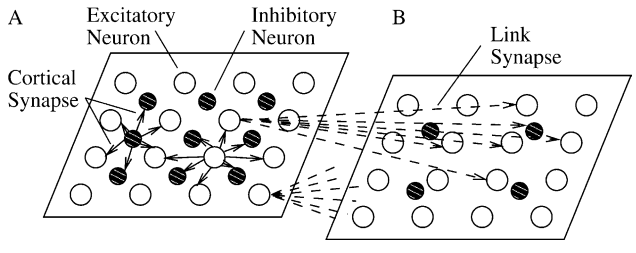

Fig. 1. Schematic representation of a computational model with two cortical agents A, B, appropriately connected by a link structure. Neighbouring neurons are connected by intra-cortical synapses (e.g. neurons in cortical agent A). Link synapses are defined among neurons of different cortical planes located at neighbouring positions (e.g. inter-cortical synapses across agents A and B).

3.2. Link agent

An appropriate link agent is specified to allow information flow across cortical agents. Using the link structure, any two cortical agents can be connected. Thus an arbitrary complex connectivity can be defined, to simulate connectivity of brain areas.

Each link agent is specified by two sets of one-way synapses. Only excitatory neurons are used as outputs of the efferent cortical agent, while both excitatory and inhibitory neurons receive input in the afferent agent (Fig. 1). Synapse definition follows the principle that neighbouring cells project to neighbouring areas. Thus, a locality threshold is also employed to specify neighbourhood across cortical agents. This is achieved by using the spatial properties inherited by the planar cortical model. Locality is approximated by the circular neighbourhoods defined after projecting the neurons of the linked cortical agents on a common virtual plane (Fig. 1). Before its projection to the common plane, the plane of afferent agent can be appropriately scaled and rotated. Thus, increased flexibility on link synapse definition is offered.

3.3. Learning rules

Epigenetic learning has an important contribution to the performance of the brain (Thelen, 2000). To enforce experience-based subjective learning of robots, each set of synapses (for both link and cortical agents) is assigned a Hebbian-like, biologically plausible learning rule, similar to (Floreano & Mondada, 1996). We have implemented a pool of 10 Hebbian-like rules that can be appropriately combined to produce a wide range of functionalities. Thus, adequate flexibility is offered to cortical and link agents to develop a desired behaviour. These rules have been selected based on their simplicity and their previous application in a variety of tasks (Choi, 2002; Floreano & Mondada, 1996; Hafner, 2000; Kohonen, 1998; Oja, 1982; Palmieri, Zhu, & Chang, 1993; Schraudolph & Sejnowski, 1992). Still, the architecture of agents is open and amenable to other learning rules with desirable characteristics in terms of either model performance or biological plausibility.

Each learning rule is specified by a unique identification number. Assuming that there is a synapse with

strength w_{ab} from neuron a with activation x_a to neuron b with activation x_b, then employed learning rules are described below:

1. *Differential decorrelation* (Choi, 2002): $\Delta w_{ab} = -\dot{x}_a \dot{x}_b$, where $\dot{x}$ is approximated by its discrete time counterpart $\dot{x}(t) = x(t) - x(t-1)$.
2. *Differential correlation* (Choi, 2002): $\Delta w_{ab} = -\dot{x}_a \dot{x}_b$, where $\dot{x}$ is similar as above.
3. *PostSynaptic* (Floreano & Mondada, 1996): $\Delta w_{ab} = w_{ab}(x_a - 1.0)x_b + (1.0 - w_{ab})x_a x_b$.
4. *PreSynaptic* (Floreano & Mondada, 1996): $\Delta w_{ab} = w_{ab}(x_b - 1.0)x_a + (1.0 - w_{ab})x_a x_b$.
5. *Covariance* (Floreano & Mondada, 1996):

$$\Delta w_{ab} = \begin{cases} (1 - w_{ab})t, & \text{if } t > 0 \\ w_{ab}t, & \text{otherwise} \end{cases}$$

where $t = \tanh(2 - 4|x_a - x_b|)$

6. Connectedness (Hafner, 2000): $\Delta w_{ab} = 1 - w_{ab}$.
7. Kohonen (1998): $\Delta w_{ab} = x_a - w_{ab}$.
8. *PCA* (Oja, 1982): $\Delta w_{ab} = x_b(x_a - x_b w_{ab})$.
9. *AntiHebbian I* (Palmieri et al., 1993): $\Delta w_{ab} = k - x_a x_b$, $k > 0$ a small forgetting factor, to avoid vanishing.
10. *AntiHebbian II* (Schraudolph & Sejnowski, 1992): $\Delta w_{ab} = k + (-2x_a x_b / x_b^2 + 1)$, where k is similar as above.

Synapse sets in all agent structures are assigned a rule, which specifies the dynamics of the epigenetic learning process. The assignment of the appropriate learning rule in each synapse set allows the emergence of the desired performance in each partial computational model, after a certain amount of robot–environment interaction.

The plasticity of agent structures, which stems from the assignment of learning rules, allows synaptic adjustments at run-time. Consequently, a large number of synapses (in the order of thousands in our case) can be self-organized based on internal agent dynamics and environmental experience. The most common, but harder to evolve, alternative of genetically encoded synaptic strengths, results to a rather unmanageable problem complexity.

3.4. Agent design specification

In the previous sections, we have presented the general models of cortical and link agents. Except of the predefined number of excitatory and inhibitory neurons of cortical agents, the other details of their structure are parametrically specified. Neural parameters (μ, α, β) are defined by six real values, separately for excitatory and inhibitory neurons. The radii used for the definition of the four synapse sets (excitatory–excitatory, excitatory–inhibitory, inhibitory–excitatory, inhibitory–inhibitory) are specified by four real values. Additionally, four integers specify the identifiers of the learning rules which adjust synapse weights during lifetime performance. In summary, 14 parameters are necessary to specify the complete structure of a cortical agent.

Synapse set definition for link agents is supported by two real values specifying the necessary inter-cortical neighbourhood radius (efferent excitatory to either excitatory or inhibitory afferent neurons). Additionally, two integers specify the identifiers of the learning rules which adjust synapse weights in each synapse set. The rotation and scaling of the afferent cortical agent plane are defined by two more real values. In total, the structure of link agents is specified by six parameters.

Similar to a phylogenetic process, the specification of parameter values for all agents is approached in a systematic way by using an evolutionary mechanism. To support the autonomy of agents, a coevolutionary method is employed with separate species for each agent structure. This is described below.

4. Two-level collaborative coevolution

The majority of applications that involve evolutionary processes employ a single solution representation to map genotypes to phenotypes. This is also the usual case for the evolution of agent structures (Landau & Picault, 2001; Lee, 2003). However, using this approach, it is not possible to sufficiently explore partial solutions, which correspond to partial specifications of the genotype (Poter & De Jong, 2000).

To alleviate for that, coevolutionary algorithms have been recently proposed that facilitate exploration, in problems consisting of many decomposable subcomponents (Casillas, Cordon, Herrera, & Marelo, 2001). They involve two or more coevolved species (populations) with interactive performance. Each species is allowed to evolve separately, by using its own evolutionary parameters (e.g. encoding, genetic operators). Accordingly, increased search competences are inherently available in coevolutionary algorithms, while the special characteristics of each species can be preserved. Most of the coevolutionary approaches presented in the literature can be classified as competitive (Rosin & Belew, 1997), or collaborative (Poter & De Jong, 2000). Competitive approaches are based on an antagonistic scenario, where the success of one species implies the failure of the other. In contrast, collaborative approaches follow a synergistic scenario, where individuals are rewarded when they successfully cooperate with individuals from the other species. Since brain modelling efforts aim at the cooperative performance of partial structures, in the following we only consider collaborative coevolution.

Despite the increased number of applications of collaborative coevolutionary algorithms, the significance of collaborator choice is usually overlooked (Wiegand et al.,

2001, 2002). The majority of existing applications consider only the ability of species to cooperate with the best individuals from the other species (Krawiec & Bhanu, 2003; Landau & Picault, 2001). Following this heuristic, evolution is driven to a direction of reduced diversity, since all individuals of one population have to cooperate with the same (best) partial solution suggested by the remaining species. Even the additional random selection of more collaborators, followed by some approaches (Casillas et al., 2001; Gomez & Miikkulainen, 1999), is not always able to improve the performance, especially in the case of more than two coevolved species. Evidently, the coevolutionary process could be supported by the maintenance of successful collaborator assemblies, as in Moriarty and Miikkulainen (1997). This can be accomplished by employing a multiple level evolutionary approach (Delgado, Von Zuben, & Gomide, 2004).

We have introduced a new evolutionary scheme (Maniadakis & Trahanias, 2004) which combines both aspects mentioned above. Besides species evolution, our method employs an additional higher-level evolutionary process, to select the proper individuals from each species that cooperatively are able to construct a good problem solution. Thus, exploration is performed concurrently in two different spaces. The lower level evolution of each species supports search in distinct partial domains of the parameter space, while at the same time, another high level evolutionary process searches within species to identify the best collaborator schemes. The higher-level evolutionary process is able to memorize good configurations of collaborating individuals across consecutive epochs. These configurations can be used as a basis to drive coevolution, since individuals are more likely to be members of good collaborator schemes. At the same time, we introduce a new genetic operator, termed Replication, which enforces exploration across species.

Moreover, in the current work we extent the coevolutionary scheme to allow different fitness functions for different species. This is particularly important for coevolution of agents since different objectives can be defined for each agent. The latter affects the global CNS model in two ways. First, it preserves the autonomy of agents which represent CNS areas. Second, by means of the fitness functions, distinct roles can be assigned to each agent, similar to the role of the respective brain area.

We note that chromosomes are designed in an abstract form, able to handle a variety of computational structures. Thus, neural agents of any level of biological plausibility can be employed to represent CNS areas. The details of the proposed coevolutionary scheme are presented below.

4.1. Encoding

A general purpose genotype is employed for both the lower level evolution of species, and the higher-level collaborator selection process (Fig. 2(a)). According to that, each individual is assigned an identification number and encodes the values of a predefined number of variables, depending on the application. There are two different kinds of variables. The first kind is allowed to get a value from a set of unordered numbers (e.g. $\{1, 5, 7, 2\}$, with the ordering of the elements being of no use). These variables are called *SetVariables* and they are employed to store identification numbers, encoding the relationship between various elements of the CNS model. The second kind of variables is allowed to get a value within a range of values (e.g. [0,1]); therefore, they are called *RangeVariables* and they are employed to search the domain of parameters in partial structures. The values of the variables are encoded in the genome by an integer and a real number, respectively, for the two kinds. They are graphically represented with the dashed and solid boxes (Fig. 2(a)).

Based on the genome structure, we have implemented crossover and mutation operators which slightly differ from the standard ones. During the mate process, the usual single-point crossover is applied separately for SetVariables and RangeVariables (Fig. 2(b)). Different mutation operators are implemented for each kind of variables. Mutation corresponds to a random assignment in the case of SetVariables and to additive noise in the case of RangeVariables (Fig. 2(c)).

4.2. Coevolutionary scheme

The evolutionary process at the higher level performs on a population of individuals, which consist only of SetVariables. The number of SetVariables is equal to the number of species at the lower level. Thus, each SetVariable is joined with one lower level species. The value of a SetVariable can be any identification number of the individuals from the species it is joined with. Coevolved species at the lower level consist of individuals which follow the genome prototype described above, with both SetVariables and RangeVariables, depending on the parametric specification of the substructure to be modelled.

A schematic representation of an evolutionary process with three coevolved species at the lower level is illustrated in Fig. 3. We assume the existence of one cortical agent and two link agents representing its afferent and efferent projections (Fig. 3(a)). One lower level species is employed to evolve each agent structure, while a top level evolutionary process searches for cooperable individuals of agents among species (Fig. 3(b)). Individuals of the top level species consist only of SetVariables encoding identifiers of individuals at the lower levels. The genotype of individuals at low level species consists of both SetVariables and RangeVariables, as it is discussed in Section 3.4. In short, for the individuals of cortical agent species, four SetVariables encode learning rule identifiers, and 10 RangeVariables encode neural parameters of excitatory and inhibitory neurons, and neighbourhood radii used for the definition of intra-cortical synapse sets. Individuals of

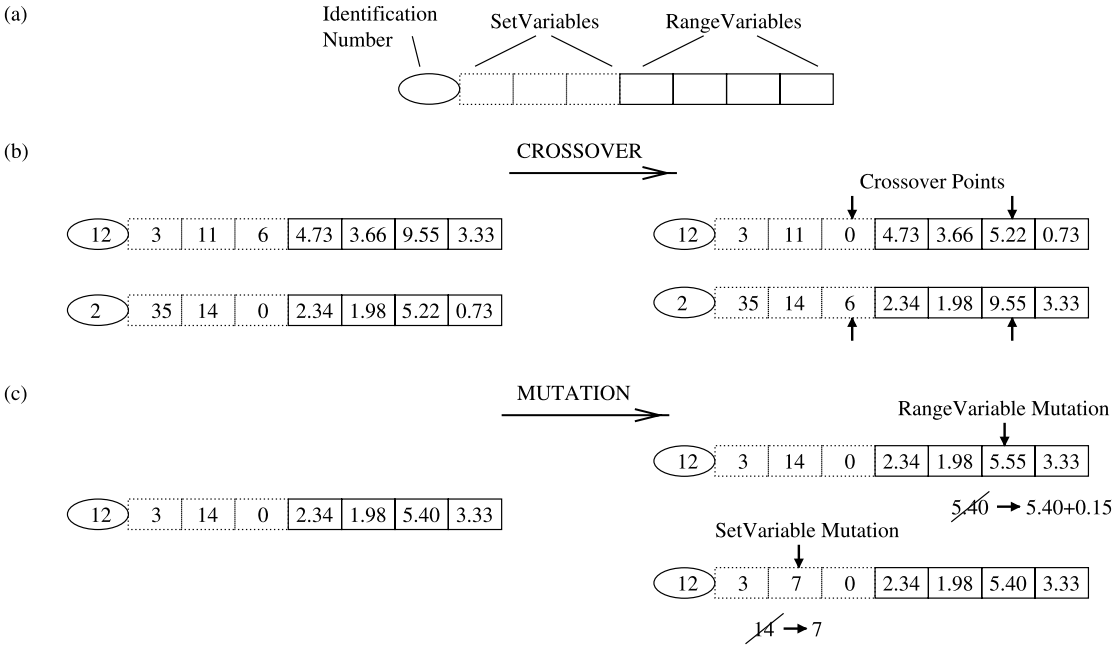

Fig. 2. A schematic representation of (a) genome structure, (b) crossover operator, (c) mutation operator.

link agent species employ two SetVariables to encode learning rule identifiers and four RangeVariables to encode rotation, scaling, and inter-cortical neighbourhood radii used for synapse definition.

In order to test the performance of individuals, the population at the higher level is sequentially accessed. The values of SetVariables at the higher level are used as guides to select collaborators among species. The collaborators are decoded to detailed agent structures. Then, agents are appropriately combined to form the proposed solution, which is further tested.

Because of the probabilistic nature of the process, some individuals of the species at the lower level could be multiply selected to participate in various combinations. Multiple collaborations are generally a drawback for the coevolutionary process. This is due to the fact that different collaborators would demand evolution of the same individual in different directions.

At the same time, some individuals in the same species might exist, which are not offered any collaboration (termed *non-collaborative* henceforth). Unused individuals can be utilised to decrease the multiplicity of collaborations for those which are heavily reused. This is achieved by introducing a new genetic operator termed 'Replication'.

For each non-collaborative individual x of a species, replication identifies the fittest individual y with more than $\max_c$ collaborations. The genome of y is then copied to x, and x is assigned $\max_c - 1$ collaborations of y, by updating the appropriate individuals of the population at the higher level. After replication, individuals x and y are allowed to evolve separately following different directions. Replication is illustrated in Fig. 4. The lower level individuals with *ids* 14, 7, 29, 9 are offered 5, 2, 0, 3 collaborations, respectively (Fig. 4(a)). Assuming that $\max_c = 3$, individual 14 is heavily reused. At the same time individual 29 is offered no collaboration at all. By applying replication, the genome of 14 is copied to 29 and two of the collaborations are appropriately redirected (Fig. 4(b)). From now on, crossover and mutation operators can separately evolve individuals 14 and 29.

The evolutionary step for the populations at all levels starts by sorting individuals according to their fitness values. Replication is applied to reduce multiple collaborations. Then, a predefined percentage of individuals are probabilistically crossed over. An individual selects its mate from the whole population, based on their accumulative probabilities. Finally, mutation is applied in a small percentage of the resulted population. Genetic operators are applied in both levels in a similar way, as described above.

4.3. Fitness assignment

Even if the majority of existing collaborative coevolutionary methods assume that all species share a common fitness function (Casillas et al., 2001; Krawiec & Bhanu, 2003; Wiegand et al., 2001), our method allows the employment of separate fitness functions for each species. This is in accordance to the coevolution of agent structures, because different objectives can be defined for each agent. Thus, the evolution of different agents is driven by the fulfilment of their own objectives, which specify their role in the global model. Additionally, the agents should also learn to perform as a group. To accomplish cooperative performance of agents, a separate fitness function for the top level evolutionary scheme is utilized to define the objective of the group. Thus the proposed coevolutionary scheme is

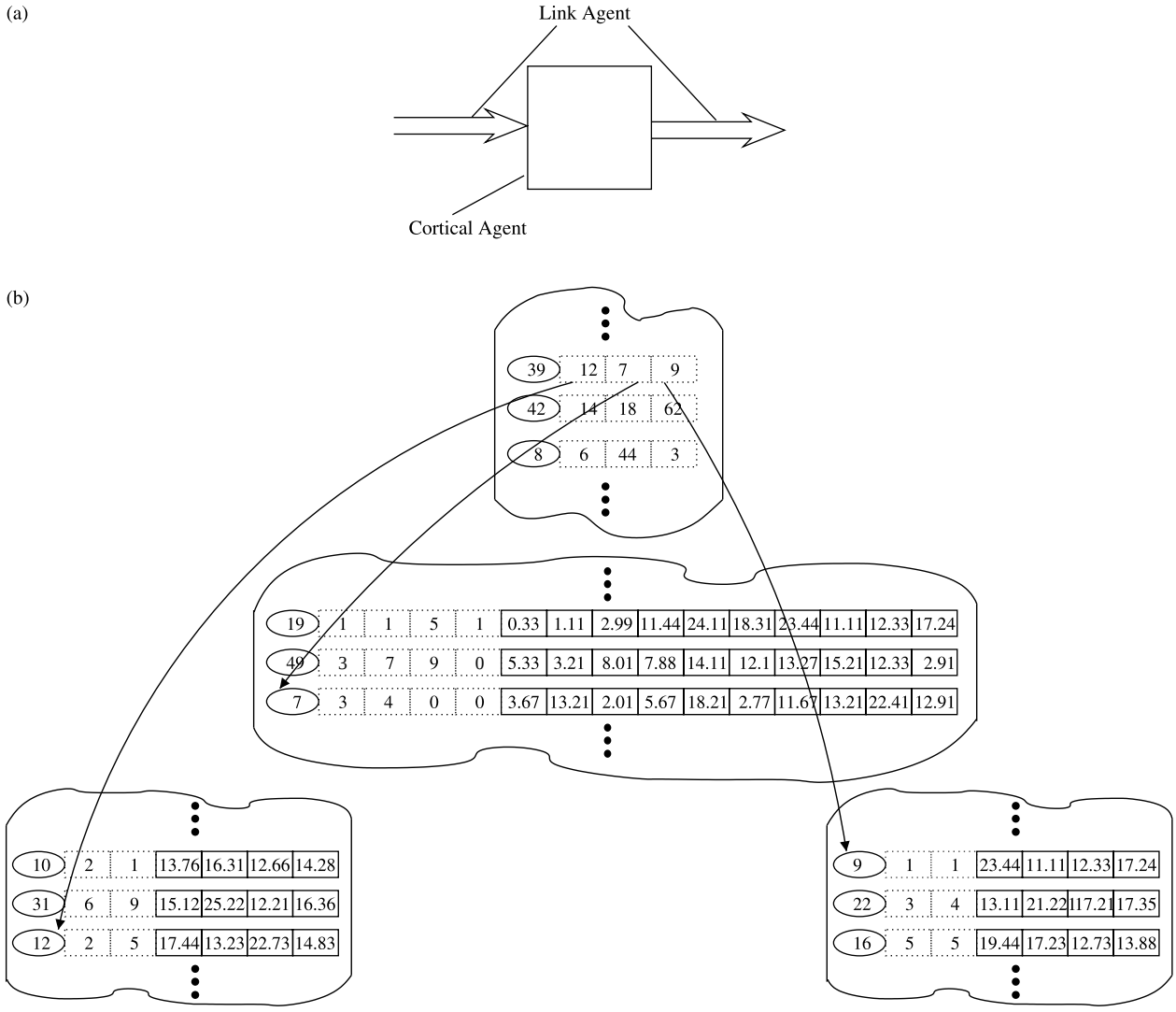

Fig. 3. Design of agents by means of collaborative coevolution. Part (a) represents schematically a hypothetical connectivity of agents. Cortical agent is illustrated with a block, while link agents are illustrated with double arrows. Part (b) illustrates the coevolutionary scheme used to evolve partial structures, with the coevolution of three low level species, tuned by a high level evolutionary process. See text for details.

able to support both the specialities of agent structures and their collaborative interactions. Both of them are particularly useful for brain modelling tasks.

When an assembly of collaborators is tested, the cooperative performance of all agent structures is evaluated. The fitness function of each species evaluates subjectively the overall performance that is it evaluates the performance according to the objectives it is designed for. Thus, for each species s, a distinct fitness value f^s is evaluated for the needs of its evolution; this evaluation is independent of the level of the species in the evolutionary process.

During fitness assignment, the individuals of the higher level are assigned a fitness value f^{TOP}, representing the appropriateness of the solution formed by the selected collaborators. Similarly to most existing approaches, individuals of the coevolved species at the lower level are assigned the maximum of the fitness value achieved by all the solutions formed with their membership. Thus an individual of the sth lower level species is assigned the value $f^s = \max_i\{f_i^s\}$, where f_i^s is the fitness value of the ith solution formed with the collaboration of the individual under discussion.

5. Experimental methodology

The effectiveness of the proposed approach is illustrated in the design of two partial brain computational models, namely motor cortex and hippocampus. The relevant experiments are indicative of the proposed coevolutionary CNS modelling approach. In the named tasks, the coevolutionary process has to perform structural and parametrical tuning in a combination of cortical and link agents, which model the performance of the respective

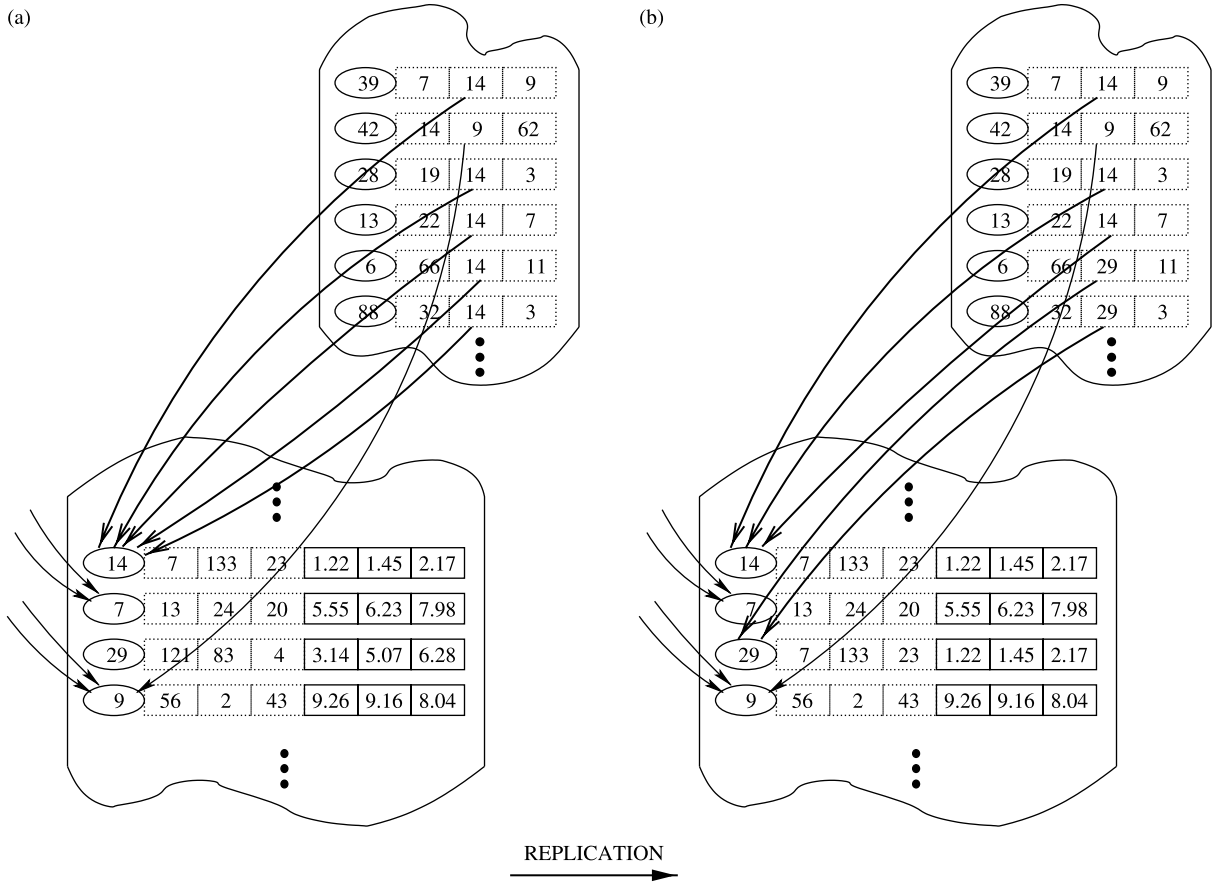

Fig. 4. Schematic representation of the replication operator ($max_c = 3$). Collaborations of individuals considered by replication operator are illustrated by thick arrows, while additional connections which are not considered by replication are illustrated by thin arrows (see text for further explanations).

brain areas. We use one species for each partial component of the computational model (either cortical or link agent). The chromosomes employed by each species are the ones described in Section 3.4.

The coevolutionary-designed models are embedded in a simulated robot to furnish it with cognitive abilities and prove the validity of the result. We employ a two wheeled robotic platform equipped with eight distance and light sensors, uniformly distributed in a circular manner around the robot, to support environmental interaction.

5.1. Modelling primary motor cortex

In the present experiment we aim at implementing a computational model of primary motor cortical areas able to emerge a wall avoidance navigation behaviour, after a certain amount of interaction with the environment. Computational models of the same brain areas have also been proposed in the literature, e.g. (Ajemian et al., 2000; Todorov, 2000), which, however, do not emphasize on the self-organized robot understanding of environmental characteristics.

Specifically, we demonstrate the implementation of a computational model of Primary Motor Cortex (M1) and Spinal Cord. The connectivity of neural network structures is illustrated in Fig. 5. Sensory stimuli is projected to the motor cortex via link agents and from there to the spinal cord with another link structure.

Following the mammalian paradigm, sensory information is somatotopically organized. Somatotopy is represented by a circular organization of environmental distance sensor. In the present experiment, we do not illustrate the projection of light stimuli in motor cortex since it does not offer valuable information for the development of wall avoidance behaviour.

Spinal cord is simulated only by its descending pathway and a cortical agent is employed to represent it. We assume the existence of an agonist and antagonist muscle in each side of a robot wheel. One motor neuron of the spinal cord

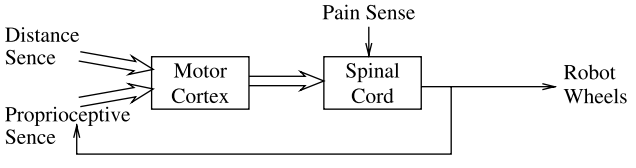

Fig. 5. A schematic overview of the Primary Motor Cortex model. Cortical agents are illustrated with blocks, while link agents are illustrated with a double arrow.

activates each of these muscles. Wheel speed is defined by the activation difference between the muscles. Thus, four motor neurons are necessary to define muscle's activation, and by consequence the speed of the robot. Proprioceptive information of muscles activation is fed back to the motor cortex, organized in a rectangular form.

Pain sensors are activated when robot bumps on the wall, and are directly projecting to spinal cord motor neurons to produce a reflexive movement. This process is very important for the early steps of learning since it prevents robot from getting stack against the wall and allows it to continue interaction with the environment.

Motor cortex agent is represented by a neural structure with 49 excitatory and 36 inhibitory neurons. Spinal cord is represented by four excitatory motor neurons with zero membrane time constant. All cortical and link agents are randomly initialised with synaptic weights close to 0.1.

The whole computational model consists of five subcomponents (two cortical and three link agents) which have to cooperate to accomplish the desired performance. A higher-level evolutionary process with genomes of five SetVariables tunes the coevolution of all five species following the method presented in Section 4. Populations of 150 individuals evolve all subcomponent species, while a population of 300 individuals evolves the higher-level collaborator selection process. In the present experiment, a common fitness function is used for both high and low levels of coevolution. Each individual is assigned a fitness value according to the function

$$f = \left(\sum_{M}(\mathrm{sl}+\mathrm{sr}-1)(1.0-p^2)\right) \left(1-\frac{2}{M}\left|\sum_{M}\frac{\mathrm{sl}-\mathrm{sr}}{\mathrm{sl}\times\mathrm{sr}}\right|\right)^3\left(1-\frac{2B}{M}\right)^3 \quad (2)$$

where we assume that robot performance is observed for M steps, sl, sr are the instant speeds of the left and right wheel, p is the maximum instant activation of distance sensors, and B is the total number of robot bumps. The first term of Eq. (2) seeks for forward movement far from the walls, the second supports straight movement without unreasonable spinning, and the last term minimizes the number of robot bumps on the walls.

The robot is allowed to interact with the environment for $2M$ simulation steps. To avoid the effect of random initialisation, robot performance is only observed for the last M steps. Thus neural agent structures are given enough time to self-organize synaptic weights and develop the desired behaviour. Fitness value is evaluated only in that period, following Eq. (2). In the present experiment, we use a number of $M=1500$ simulation steps.

Evolution was performed in synchronous steps for all populations with 0.7 crossover probability and 0.08 mutation probability. After 50 epochs we got many computational structures able to drive the robot without bumping on the walls. The total amount of synapses that the coevolutionary process specified for both cortical and link agents varies from 1100 to 1450 synapses. A sample result is illustrated in Fig. 6.

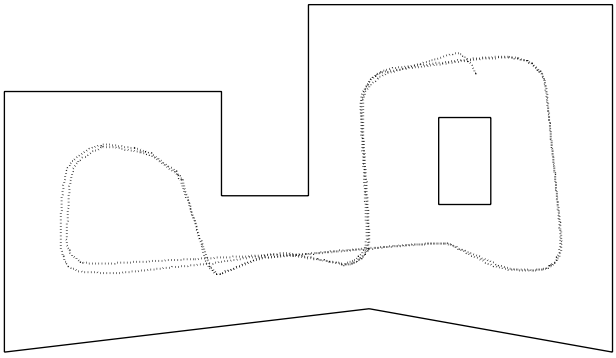

Fig. 6. A sample result of robot wandering navigation.

5.2. Modelling hippocampal formation

The hippocampus is one of the most studied areas of the mammalian cortex because of its prominent role in the memorization of spatial information. Different groups of cells, namely place cells, have been detected in the mammalian's hippocampus, which preferably fire when the animal is in a particular portion of its environment, but they are largely independent of its orientation and actual view (O'Keefe, 1976). The hippocampus consists of partial areas which cooperate to develop place cells. Thus, the proposed model does not employ a single module to model the hippocampal performance, even if it could have been possible for the present experiment. The reason is that it is far from the biological hippocampal connectivity, and it will be not able to reproduce any additional hippocampal data. Following recent trends in the area, we focus our study in the investigation of the entorinal cortex (EC) from parahippocampal region and dentate gyrus (DG), and Amon's horn structures CA3, CA1 from hippocampal formation. Recently, place cells have been detected in all these structures. Since the exact role of hippocampal areas has not been specified yet in the literature, the design of the computational model will be based on existing knowledge, that is development of place cells in hippocampal substructures.

A number of hippocampal computational models have been proposed in the literature, which are able to develop place cells based on allothetic sensory stimuli. Some approaches consist of an arrangement of appropriately connected neurons on a planar map (Arleo & Gerstner, 2000; Hafner, 2000). Other hippocampal models are based on the recurrent connectivity of CA3 neurons (Kali & Dayan, 2000). A combination of planar map with recurrent connections is presented in Samsonovich and McNaughton (1997). However, according to Eichenbaum, Dudchenko, Wood, Shapiro, and Tanila (1999)

and O'Keefe, Burgess, Donnett, Jeffery, and Maguire (1998), the existence of a topographical relation between environmental location and hippocampal cells seems not valid. This is taken into account in Stringer et al. (2004), where attractor networks are employed to perform feature encoding. Additionally, the majority of existing models employ simplified structures which omit the projection from CA1 to EC. This is a very critical design decision, since a recurrent cellular structure is computationally represented by a feed forward one. A computational model with re-entrant projections from CA1 to EC is presented in (Norman & O'Reilly, 2001), but it is not tested for the development of place cells.

In the present experiment, we present a detailed hippocampal model with separate neural agents representing each hippocampal area (EC, DG, CA3, CA1). Thus, all interactions within these areas can be simulated. Similar to the majority of the models, we follow an approach based on environmental features for the development of place cells, but in contrast to them we do not assume global view of the environment. Appropriate fitness functions drive the coevolutionary process (as explained below), aiming at the development of place cells.

It has been experimentally shown that the hippocampal system processes allocentric (orientation invariant) information (Burgess, Becker, King, & O'Keefe, 2001). This is a common hypothesis for all computational models. We have implemented a simple computational formula to perform this transformation, given the current orientation ϕ of the animal (Maniadakis & Trahanias, 2003). For the sake of simplicity we assume that the number of head-direction (HD) neurons is equal to the number of light or distance sensors; let this number be M. Each HD neuron has a preferred direction θ of maximal activation and follows the gaussian model, similar to real HD cells (Taube, 1998). Let us assume that the information of the ith egocentric sensor is given by h_i. The allocentric measure is achieved by the following summation over all HD neurons

$$f_i = \frac{\sum_{j=0\ldots M-1} e^{-(\phi-\theta_{(M-j)\bmod M})^2} h_{(i+j)\bmod M}}{\sum_{j=0\ldots M-1} e^{-(\phi-\theta_{(M-j)\bmod M})^2}} \quad (3)$$

where f_i is the new orientation invariant measure. This formula has a slight smoothing effect in sensory stimuli, which is due to the averaging performed. Intuitively, it considers stimuli from all sensors, rotated by certain angles, and weighted each time by a factor that is proportional to the matching of rotation and head direction. It is interesting to observe that this formula can be directly used to combine our approach with other computational models that develop HD cells (e.g. Redish et al., 1996).

Hippocampal model is fed with allocentric measures. Both senses (i.e. distance and light) are somatotopically organized following a circular representation. Distance and light sensory stimuli are projected to EC and then they travel along partial hippocampal structures as it is illustrated in Fig. 7.

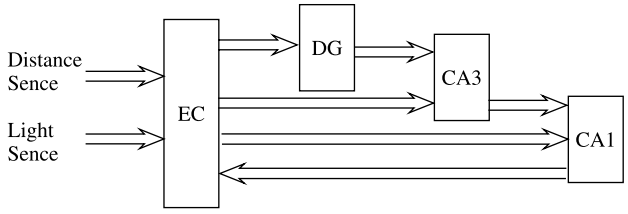

Fig. 7. A schematic overview of the hippocampal model. Cortical agents are illustrated with blocks, while link agents are illustrated with a double arrow.

The complete computational model consists of 12 subcomponents (four cortical and eight link agents) which have to cooperate to accomplish place cell development. The distribution of neurons in cortical agents is in analogy with the biological prototype, as it is shown in Table 1. The coevolutionary scheme presented in Section 4 is employed to perform design decisions for all computational structures. A higher-level evolutionary process with genomes of 12 SetVariables tunes the coevolution of all 12 species. Each agent species evolves 150 individuals, while the higher-level collaborator selection process evolves a population of 320 individuals.

In order to test the development of place cells, we define P ($P=13$ in this experiment) areas in the environment (see Fig. 8), where the activation of hippocampal excitatory neurons is observed. The activation of inhibitory neurons is not examined, since only excitatory neurons encode efferent information. For each cortical agent $i\in\{\text{EC, DG, CA3, CA1}\}$, and each location $p\in\{1,\ldots,P\}$, separate activation-averages over time, a_j^{ip}, are computed, with j identifying excitatory neurons.

Place cell development implies that when the robot is positioned in two different areas, the rate rd of differences within activation-averages, divided by the total activation, should be close to one. For two locations p, q, with $q\neq p$, this measure is expressed mathematically by:

$$rd^i(p,q) = \frac{\sum_j |a_j^{ip} - a_j^{iq}|}{\sum_j (a_j^{ip} + a_j^{iq})} \quad (4)$$

A successful development of place cells in cortical agent i implies that the average activations at any two

Table 1
The distribution of neurons in hippocampus

Area	Rat	Model
EC	200,000	85 (Exc: 49, Inh: 36)
DG	1,000,000	130 (Exc: 81, Inh: 49)
CA3	160,000	61 (Exc: 36, Inh: 25)
CA1	250,000	100 (Exc: 64, Inh: 36)

The second column presents a rough estimate of neurons in rat hippocampal areas (Norman & O'Reilly, 2001). The number of excitatory and inhibitory neurons used in each cortical agent is presented in the third column.

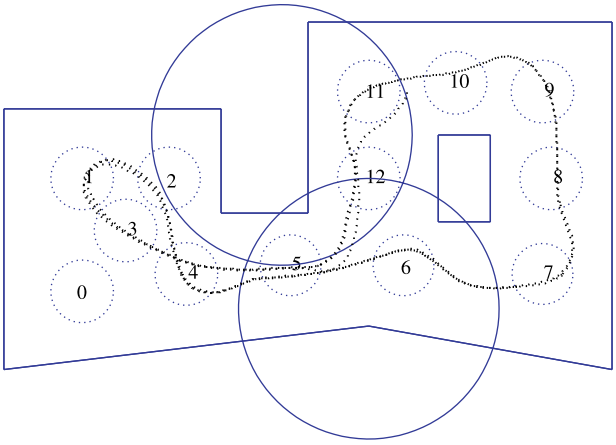

Fig. 8. Experimental setup of the hippocampal model testing. Environmental areas ($P=13$) are illustrated with dashed circles. Solid circles illustrates the existence of two light sources.

locations p, q give high values of relative difference rd. Following a worst case scenario, a separability measure of place cells RD^i can be defined based on the most similar locations:

$$RD^i = \min_{p,q}\{rd^i(p,q)\} \tag{5}$$

Another measure is also defined to support the stability of cortical agents performance. It estimates the consistency of activations for the case that the robot is located in area p more than once. This can be done using the contrast of activation-average values at p. We use the variance v^{ip} as a contrast measure:

$$v^{ip}\frac{1}{N_e}\sum_j |m - a_j^{ip}|, \quad \text{with } m = \frac{1}{N_e}\sum_j a_j^{ip} \tag{6}$$

for a cortical agent i, with N_e excitatory neurons. If the same neurons are activated every time the robot is located in p, contrast measure v^{ip} will have a high value, while if different neurons are activated every time the robot is located at the same p, then v^{ip} will have a small value. The average of variances at all locations is employed as a consistency measure of cortical agent i:

$$V^i = \frac{1}{P}\sum_p v^{ip} \tag{7}$$

Since CA1 is the major efferent structure of hippocampus, we assume that the excitatory activation in CA1 agent should be able to infer the location of the robot. This is done at every simulation step by estimating the distance of current activation x_j^{CA1} with the activation-averages of CA1, at every location p:

$$d^p = \sum_j \left|x_j^{CA1} - a_j^{CA1_p}\right| \tag{8}$$

A simple process infers the robot location p as the one with the minimum distance d^p.

Using the location inference process described above, a success rate $S=s/t$ is defined for the total hippocampal model. It is based on the number of simulation steps s that the inference process is able to successfully identify the robot location, relative to the total number of simulation steps t that the robot is located in known positions.

The fitness function which supports the coevolutionary process is based on the measures described above. Since different brain areas exhibit different functionalities, it is necessary to define separate objectives for each partial structure. Fortunately, following an agent-based brain modelling, different objectives can be defined for each partial area. The coevolutionary approach described in Section 4 is particularly suitable to support this task, due to the fact that separate fitness functions can drive the evolution of each species.

Since the exact functionality of hippocampal substructures is a matter of current research, the coevolutionary method exploits only existing knowledge concerning the development of place cells in all substructures, to implement an appropriate hippocampal model. Thus, it is implicitly assumed that the objective of partial agents is the development of their own place cells. Additionally, since the connectivity of substructures follows the biological prototype and the coevolutionary method inherently supports their cooperation, partial structures develop a functionality similar to the respective brain areas.

The fitness function F^i employed for the evolution of the ith cortical agent species is defined by:

$$F^i = RD^i\sqrt{V^i}S \tag{9}$$

(e.g. $F^{EC} = RD^{EC}\sqrt{V^{EC}}S$, for the EC cortical agent). The first term seeks for increased separability of place cells in the respective partial hippocampal area, the second term supports the consistency of place cell firing, and the third maximizes the success rate of the overall hippocampal model.

The fitness function $F^{i\Rightarrow j}$ employed for the evolution of the link agent that supports efferent projection from cortical agent i to cortical agent j, follows a similar pattern. It is defined by:

$$F^{i\Rightarrow j} = RD^j\sqrt{V^j}S \tag{10}$$

(e.g. $F^{EC\Rightarrow DG} = RD^{DG}\sqrt{V^{DG}}S$, for the link agent connecting EC to DG). Since link agents are consisting of one way synapses, they only affect the performance of the afferent module. Intuitively, Eq. (10) assumes that cortical agent i functions properly, and the (one way) link agent $i\Rightarrow j$ is designed to support the performance of cortical agent j.

The higher-level evolutionary process is driven by the fitness function $F^{TOP} = RD^{CA1}\sqrt{V^{CA1}}S$. This is equal to the fitness function of CA1 cortical agent because this structure is the major output gateway of hippocampus.

Evolutionary learning is performed following a two phase incremental procedure. In the first phase (epochs 1–20)

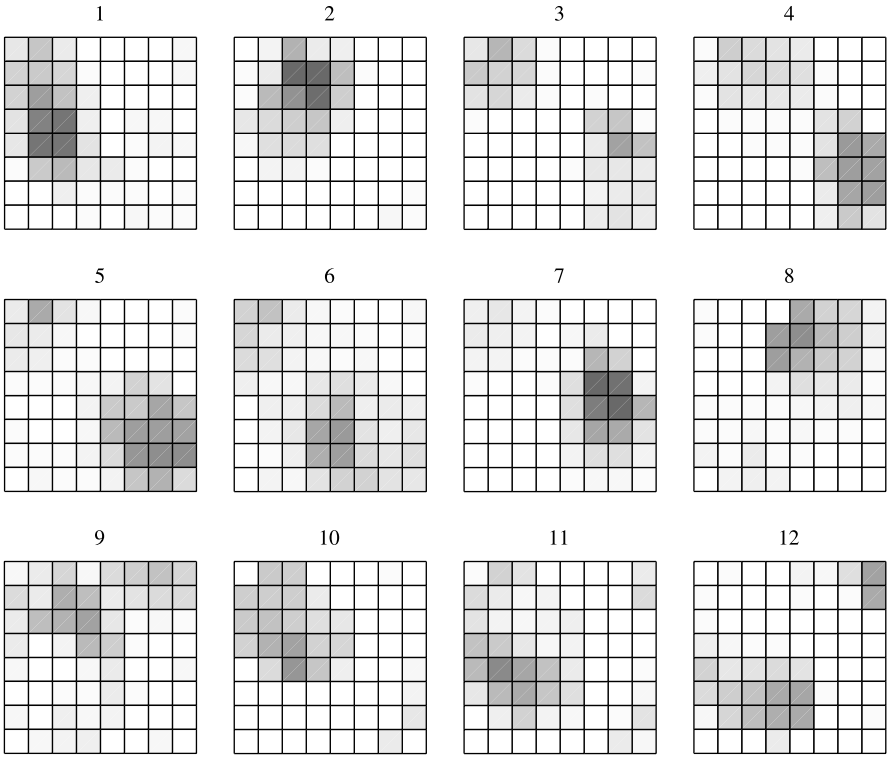

Fig. 9. Development of place cells at the CA1 agent for the environment areas of Fig. 8. Activation of cells is illustrated with levels of grey.

the robot is randomly moved to one of the P areas every k simulation steps ($k=25$ for the results illustrated here). Hippocampal processing is then performed with a standing robot. This phase enforces the fast consideration of environmental differences at various locations, by the first generations of hippocampal structures. In the second phase (all following epochs), the random movement of the robot is stopped, and coevolutionary testing is performed with a freely moving robot, by employing the navigation behaviour described in Section 5.1. This phase additionally enforces the synchronization of the robot's wheel speed, with the change rate of activations at hippocampal neurons.

The end of the evolutionary process specified cortical and link agents with a total number of synapses ranging between 16,200 and 18,400. The results of place cell development at CA1 for the robot path of Fig. 8 are illustrated in Fig. 9. Similar to biological place cells, neural activation is able to specify environmental areas. Place cells are able to successfully infer the position of the robot, with a success rate $S=94.5\%$. The small error rate is mostly due to the phenomenon of spatial hysteresis in the activation of place cells. This is a side effect of the recurrent connectivity in hippocampal structures, which preserves very recent environmental experience, and consequently delays stabilization of place cells (Doboli, Minai, Best, & White, 2001). We can easily observe that there is no spatial relationship within the developed place cells, as it is suggested by biological studies (Eichenbaum et al., 1999; O'Keefe et al., 1998).

Finally, it should be mentioned that place cells have also been developed in all other hippocampal sub-structures (EC, DG, CA3). This is a result of the appropriate fitness functions employed for the evolution of partial agents (Eqs. (9) and (10)). In another set of experiments, we used the very same fitness function (equal to F^{TOP}) for the evolution of all species. In a qualitative comparison to the approach described in detail above, these experiments showed that the process converged successfully, developing place cells in CA1, but no place cells were developed in the remaining hippocampal sub-structures (since no such objective were described by the employed fitness function). This point indicates that the employment of a separate fitness function describing the objective of each partial agent is necessary for the successful modelling of hippocampal formation. Additionally, in the set of experiments not presented here, the speed of convergence of the evolutionary process was significantly reduced. This point denotes that the successful development of place cells in one agent assists the emergence of the desired performance in the rest partial structures.

5.3. Discussion

In both experiments described above, there were many different assemblies of cortical and link agents with successful performance. The variety of partial solutions is appropriately combined by the higher-level collaborator selection process to construct successful global solutions.

The pluralism of results offers the opportunity to developmentally add more components to represent additional models of brain areas, which will be further coevolved to obtain successful performance in even more complex tasks.

The coevolutionary process specifies a large number of synapses for both computational models (on average 1400 for a cortical agent and 450 for a link agent). This fact highlights the need for a compact encoding of agent structures, since the alternative choice of detailed genetically encoded synaptic weights would be very difficult to evolve successfully. Following the proposed agent representation with the assignment of a learning rule to each synapse set, we are able to achieve both the evolvability of agents and the self-organization of their structure.

The coexistence of different rules in the same model increases the behavioural abilities of the neural agents, but at the same time complicates their systematic analysis. The most common analysis methods address investigation of the real-time dynamics that a single learning rule imposes on agents (Vegas & Zufiria, 2004). The coexistence of several different rules in the same model (14 for motor cortex, 32 for hippocampus)[2] prevents us from performing a similar analysis, since typically in the bibliography only one rule is used for the whole model. For example rule 8 is well known to perform principal component analysis, while rule 7 is known to perform classification, but their coexistence in the same network has not a known interpretation. We also mention that it is difficult to perform even manual tests to reasonably investigate the impact of different rules at various synapses. This is because the change of only one rule may drastically affect the performance of the whole model, as it is also mentioned in previous studies (Floreano & Urzelai, 2000; Rolls & Stringer, 2000).

Therefore, only intuitive explanations can be given to the final rule choice. For example, in motor cortex experiments, anti-hebbian rules are usually selected on either the projection of sensory information to M1, or to the excitatory–excitatory synapses within M1. The first case may be explained by the need to increase M1 sensitivity when distance sensors are not able to sense the wall. During that 'no sensation' period, the synapse weight is increased by the anti-hebbian rule, and the model is prepared to react when a wall is sensed again in the future. The second case can be explained by the relationship of anti-hebbian learning to mean square error minimization (Wang, Kuo, & Principe, 1995), where the M1 agent aims at developing the appropriate sensor-to-actuator mapping. Obviously, such intuitive explanations contribute to our understanding of the modelling process, but are far from a systematic analysis, and further research is necessary in that direction.

The design of fitness functions is very crucial for the success of the coevolutionary process, since they constitute the means to assign roles in each agent structure. Their formulation can be based on either external or internal characteristics of robot behaviour (Floreano & Urzelai, 2000). External measures are those that cannot be directly estimated by the robot, but only by an external observer (e.g. distance from a non-visible position). Internal measures are those that can be directly estimated by the artificial organism (e.g. wheel speed difference). Even if the former category of measures simplifies the design of fitness functions, they are environment-specific, and offer less generalization to different circumstances. In contrast, fitness functions based on internal characteristics are usually slightly more complex, but tend to be more robust under various environmental conditions. Consequently, the fitness functions employed in the present work are based on internal measures to support the reliability of the evolutionary process.

The proposed coevolutionary approach allows separate fitness functions to be employed for the evolution of each species. This is in contrast to a unimodal evolutionary process that calls for a single fitness function, preventing the consideration of each agent's own performance. Moreover, with our approach, autonomous partial structures with collaborative performance can be designed. The combination of autonomy and collaborative performance in a single method seems particularly appropriate for brain modelling tasks.

In the first experiment described, all species employ the same fitness function. The experiment of hippocampal modelling demonstrates the power of the coevolutionary agent-based design, since it allows the definition of distinct objectives for different agents. Each agent is evolved aiming at the reproduction of the respective biological data (development of its own place cells). Consequently, in contrast to existing computational hippocampal structures, our hippocampal model is the only one with the ability to develop place cells without spatial relationship in all partial areas. The ability of the model to reproduce biological data, together with the connectivity of its subcomponents that follows the biological prototype, enforce partial structures to develop a role similar to that of the respective brain areas.

It should be noted that both results obtained are biologically plausible, but additional experiments are necessary to support their biological reliability. For example, the motor cortex model should be also employed to achieve goal directed robot motion, while the hippocampal model should also reproduce data from additional episodic memory tasks. The more biological data the model is able to reproduce, the more reliable become the emergent roles of agents. Thus, additional experiments should be designed to run in parallel with those discussed in the present study, and fitness functions should be appropriately

[2] In the case of the primary motor cortex model, 14 different learning rules are employed. This results as follows: four rules for each of the two cortical agents and two rules for each of the link agents. For the hippocampus model, 32 different learning rules are employed, four rules for each of the four cortical agents and two rules for each of the eight link agents.

modified, also aiming at the reproduction of the new data. The coevolutionary approach introduced here may be employed to search the parameter spaces of the motor cortex and the hippocampal models to specify those values which are able to reproduce all biological data simultaneously. Since in the present study we obtained very different results with successful performance, we expect that the coevolutionary method will be able to identify parameter values with the desired performance in the proposed set of experiments. This constitutes a direction of our future work.

The proposed two level coevolutionary strategy can be easily extended to a multiple level hierarchical coevolutionary strategy, which fits to large scale brain modelling tasks. Following a hierarchical coevolutionary approach, the integration of existing computational models representing different brain areas can be achieved by introducing an arbitrary number of higher-level evolutionary processes. This consists another direction of our future work. More specifically, the performance of motor cortex and hippocampal models should be combined to accomplish tasks of purposeful motion.

It should also be noted that the compound brain model does not have to perform in a hierarchical mode. The performance of partial brain structures can be either hierarchical or completely parallel. Hence, the hierarchical coevolutionary approach does not impose any further constraints. It is introduced only to support the design process of incremental brain modelling.

Finally, it is worth mentioning that the coevolutionary design method is currently enhanced, to support simulation of partial brain area lesion studies. This ability is a clear advantage offered by the coevolutionary approach. In short, appropriate partial agents are deactivated, and the remaining structures are appropriately designed to achieve the behaviour suggested by the biological prototype. Very different fitness functions specify the role of each partial structure in the complete or the eliminated model. More specifically, we are currently extending the model of primary motor cortex to achieve more complex behaviours. An additional cortical agent is employed simulating premotor areas, to modulate the performance of primary motor cortex. By following the coevolutionary approach, the entire model (both primary motor and premotor structures) is designed to develop a goal following behaviour, while the eliminated model (primary motor structure only) is designed to perform wall avoidance. Thus, the coevolutionary method is able to design CNS computational models which reproduce pre- and post-lesion performance of biological organisms. Consequently, the reliability of the model is further supported. In the near future, we aim at employing the coevolutionary approach to reproduce biological data regarding hippocampal lesions. These experiments could not be performed in the present study, since they demand more complex navigation behaviours than wall-avoidance, including planning abilities. Similar experiments are of great importance since they are able to highlight the specific role of partial structures in the global model, and also support predictions regarding lesion experiments in biological organisms.

6. Conclusions

In the present work we proposed a novel computational framework for the design and implementation of partial brain models following a coevolutionary agent-based approach. These models are embedded in a robotic platform to furnish it with cognitive capabilities. Although the biological reliability of implemented CNS models was not the focus of this paper, our method can also address this issue by additional constraints on the evolutionary process.

With regards to the implemented models, the characteristics of the employed agent based approach support partial structure autonomy. Appropriate neural agents are employed to represent brain areas following a similar connectivity to the mammalian CNS. We introduce the utilization of collaborative coevolutionary algorithms to support design specification of agent structures. The proposed coevolutionary method is suitable for the design of agents because it offers increased search abilities of partial components, and is able to emphasize both the speciality of brain areas and their cooperative performance. By following a coevolutionary method for design specification, our approach is inherently furnished with the ability to integrate partial brain models.

The work presented here constitutes a first attempt towards a rigorous method for brain modelling, based on collaborative coevolution. The results obtained attest to its validity and effectiveness in modelling partial brain areas and replicating biological behaviours. Further work is necessary and currently underway, to investigate the suitability of our approach in large-scale modelling tasks and the respective endowment of cognitive abilities to artificial systems.

References

Ajemian, R., Bullock, D., & Grossberg, S. (2000). A model of movement coordinates in motor cortex: Posture-dependent changes in the gain and direction of single cell tuning curves. Technical Report, Department of Cognitive and Neural Systems, Boston University.

Arleo, A., & Gerstner, W. (2000). Spatial cognition and neuro-mimetic navigation: A model of hippocampal place cell activity. *Biological Cybernetics*, *83*, 287–299.

Burgess, N., Becker, S., King, J., & O'Keefe, J. (2001). Memory for events and their spatial context: Models and experiments. *Philosophical Transactions of the Royal Society*, *356*, 1493–1503.

Casillas, J., Cordon, O., Herrera, F., & Marelo, J. J. (2001). Cooperative coevolution for learning fuzzy rule-based systems. *Proceedings of EA-01*.

Choi, S. (2002). Adaptive differential decorrelation: A natural gradient algorithm. *Proceedings of ICANN*.

Cotterill, R. M. J. (2001). Cooperation of the basal ganglia cerebellum, sensory cerebrum, and hippocampus: Possible implications for cognition, consciousness, intelligence and creativity. *Progress in Neurobiology*, *64*, 1–33.

Delgado, M. R., Von Zuben, F., & Gomide, F. (2004). Coevolutionary genetic fuzzy systems: A hierarchical collaborative approach. *Fuzzy Sets and Systems*, *141*, 89–106.

Doboli, S., Minai, A. A., Best, P. J., & White, A. M. (2001). An attractor model for hippocampal place cell hysteresis. *Neurocomputing*, *38–40*, 1185–1191.

Duchaine, B., Cosnides, L., & Tooby, J. (2001). Evolutionary psychology and the brain. *Current Opinion in Neurobiology*, *11*, 225–230.

Eichenbaum, H., Dudchenko, P., Wood, E., Shapiro, M., & Tanila, H. (1999). The hippocampus, memory and place cells: Is it a spatial memory or a memory space? *Neuron*, *23*, 209–226.

Floreano, D., & Mondada, F. (1996). Evolution of plastic neurocontrollers for situated agents. *Proceedings of SAB-96* .

Floreano, D., & Urzelai, J. (2000). Evolutionary robots with on-line self-organization and behavioral fitness. *Neural Networks*, *13*, 431–443.

Franklin, S., & Graesser, A. (1996). Is it an agent, or just a program? A taxonomy for autonomous agents. *Proceedings of International Workshop on Agent Theories, Architectures, and Languages* .

Geary, D. C., & Huffman, K. J. (2002). Brain and cognitive evolution: Forms of modularity and functions of mind. *Psychological Bulletin*, *128*, 667–698.

Gomez, F. J., & Miikkulainen, R. (1999). Solving non-Markovian control tasks with neuro-evolution. *Proceedings of IJCAI-99* .

Hafner, V. V. (2000). Learning places in newly explored environments. *Proceedings of SAB2000*.

Jennings, N. R. (2000). On agent based software engineering. *Artificial Intelligence*, *117*, 277–296.

Kali, S., & Dayan, P. (2000). The involvement of recurrent connections in area CA3 in establishing the properties of place fields: A model. *Journal of Neuroscience*, *20*(19), 7463–7477.

Kandel, E. R., Schwartz, J. H., & Jessell, T. M. (2000). *Principles of neural science* (3rd ed). New York: Appleton & Lange.

Klein, S. B., Cosmides, L., Tooby, J., & Chance, S. (2002). Decisions and the evolution of memory, multiple systems, multiple functions. *Psychological Review*, *109*, 306–329.

Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, *21*, 1–6.

Krawiec, K., & Bhanu, B. (2003). Coevolution and linear genetic programming for visual learning. *Proceedings of GECCO-03*.

Landau, S., & Picault, S. (2001). Modelling adaptive multi-agent systems inspired by developmental biology. *Proceedings of MASA*.

Lee, M. (2003). Evolution of behaviours in autonomous robot using artificial neural network and genetic algorithm. *Information Sciences*, *155*, 43–60.

Maniadakis, M., & Trahanias, P. (2003). A computational model of neocortical–hippocampal cooperation and its application to self-localization. *Proceedings of ECAL-03*.

Maniadakis, M., & Trahanias, P. (2004). Evolution tunes coevolution: Modelling robot cognition mechanisms. *Proceedings of GECCO-04*.

Moriarty, D. E., & Miikkulainen, R. (1997). Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation*, *5*(4), 373–399.

Norman, K. A., & O'Reilly, R. C. (2001). Modelling hippocampal and neocortical contributions to recognition memory: A complementary learning systems approach. University of Colorado, Boulder, ICS. Technical Report 01-02.

O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat. *Experimental Neurology*, *51*, 78–109.

O'Keefe, J., Burgess, N., Donnett, J., Jeffery, K., & Maguire, E. (1998). Place cells, navigational accuracy, and the human hippocampus. *Philosophical Transactions of the Royal Society*, *353*, 1333–1340.

Oja, E. (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, *15*, 267–273.

Palmieri, F., Zhu, J., & Chang, C. (1993). Anti-Hebbian learning in topologically constrained linear networks: A tutorial. *IEEE Transactions on Neural Networks*, *4*, 748–761.

Poter, M., & De Jong, K. (2000). Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation*, *8*, 1–29.

Redish, A. D., Elga, A. N., & Touretzky, D. S. (1996). A coupled attractor model of the rodent head direction system. *Computation in Neural Systems*, *7*, 671–685.

Rolls, E. T., & Stringer, S. M. (2000). On the design of neural networks in the brain by genetic evolution. *Progress in Neurobiology*, *61*, 557–579.

Rosin, C. D., & Belew, R. K. (1997). New methods for competitive coevolution. *Evolutionary Computation*, *5*, 1–29.

Samsonovich, A., & McNaughton, B. L. (1997). Path integration and cognitive mapping in a continuous attractor neural network model. *Journal of Neuroscience*, *17*(15), 5900–5920.

Scassellati, B. (2000). Theory of mind for a humanoid robot. *IEEE/RSJ International Conference on Humanoid Robotics*.

Schraudolph, N. N., & Sejnowski, T. J. (1992). Competitive anti-hebbian learning of invariants. *Advances in Neural Information Processing Systems*, *4*, 1017–1024.

Shin, J. (2002). Towards computational and robotic modelling of animal cognition and behaviour. *Neurocomputing*, *44–46*, 985–992.

Smith, J. M. (1987). When learning guides evolution. *Nature*, *329*, 761–762.

Stringer, S. M., Rolls, E. T., & Trappenberg, T. P. (2004). Self-organising continuous attractor networks with multiple activity packets, and the representation of space. *Neural Networks*, *17*, 5–27.

Taube, J. S. (1998). Head direction cells and the neuropsychological basis for a sense of direction. *Progress in Neurobiology*, *55*, 225–256.

Tempesti, G., Roggen, D., Sanchez, E., & Thoma, Y. (2002). A POEtic architecture for bio-inspired hardware. *Proceedings of eighth international conference on the simulation and synthesis of living systems, ALIFE VIII*.

Thelen, E. (2000). Motor development as foundation and future of developmental psychology. *International Journal of Behavioural Development*, *24*, 385–397.

Tkaczyk, E. (2001). Pressure hallucinations and patterns in the brain. Morehead El. *Journal of Applicable Mathematics*, *1*, 1–26.

Todorov, E. (2000). Direct cortical control of muscle activation in voluntary arm movements: A model. *Nature Neuroscience*, *3*, 391–398.

Vegas, J. M., & Zufiria, P. J. (2004). Generalized neural networks for spectral analysis: Dynamics and Liapunov functions. *Neural Networks*, *17*, 233–245.

Wang, C., Kuo, J., & Principe, J. (1995). A relation between hebbian and mse learning. *Proceedings of ICASSP '95*.

Wermter, S., & Sun, R. (2000). *Hybrid neural systems*. Heidelberg: Springer.

Wiegand, R. P., Liles, W. C., & De Jong, K. A. (2001). An empirical analysis of collaboration methods in cooperative coevolutionary algorithms. *Proceedings of GECCO-01*.

Wiegand, R. P., Liles, W. C., & De Jong, K. A. (2002). The effects of representational bias on collaboration methods in cooperative coevolution. *Proceedings of PPSN-02*.