Co-development of linguistic and behavioural skills: compositional semantics and behaviour generalisation

Elio Tuci*, Tomassino Ferrauto, Gianluca Massera, and Stefano Nolfi

ISTC-CNR, Via San Martino della Battaglia n. 44, 00185 Rome, Italy
{elio.tuci,tomassino.ferrauto,gianluca.massera,stefano.nolfi}@istc.cnr.it
http://laral.istc.cnr.it/

Abstract. Population of simulated agents controlled by dynamical neural networks are trained by artificial evolution to access linguistic instructions and to execute them by indicating, touching or moving specific target objects. During training the agent experiences only a subset of all object/action pairs. During post-evaluation, some of the successful agents proved to be able to access and execute also linguistic instructions not experienced during training. This is owe to the development of a semantic space, grounded on the sensory motor capability of the agent and organised in a systematised way in order to facilitate linguistic compositionality and behavioural generalisation.

Key words: Grounding, CTRNNs, Artificial Evolution

1 Introduction

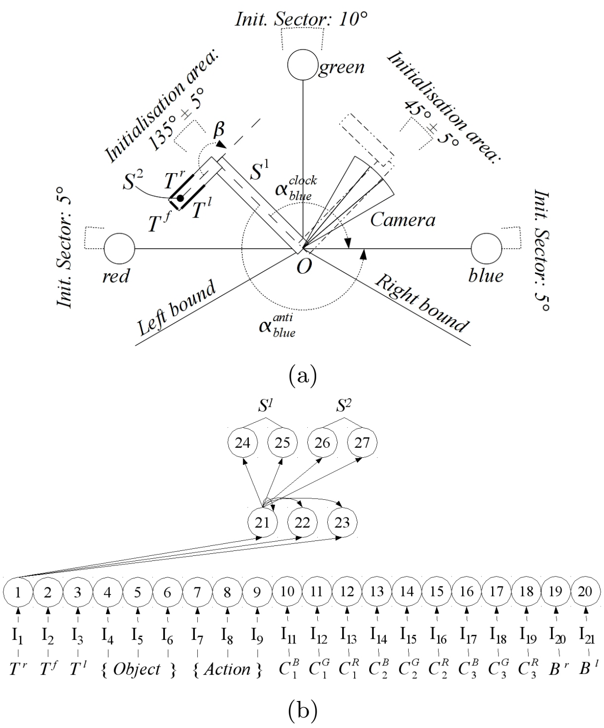
During the last few years, several researchers have been building robotic and simulated systems in which communication and linguistic skills are grounded in perception and action [1–4]. One reason that explains the interest in these works is constituted by the fact that they represent a suitable methodology to investigate with precise operational models important aspects of cognition and action [5–8]. This work is motivated by an intention to contribute to deepen our understanding of the relation between action and language in order to verify the nature of their strict interdependence. Indeed, as we will see, the results of this type of research can help us to answer important questions such as: how agents linguistic abilities are dependent on, and grounded in, other behaviours and skills; how action-language interaction supports the bootstrapping of the agents cognitive system, e.g. through the transfer of properties of action knowledge to that of linguistic representations (and vice versa).

In this paper, we describe a model in which a simulated agent interacts with coloured objects located in its peripersonal space by exhibiting three behaviours (indicating, touching, and pushing) during a series of trials. In each trial, the

* This research work was supported by the *ITALK* project (EU, ICT, Cognitive Systems and Robotics Integrating Project, grant n° 214668).

agent receives as input a linguistic instruction (constituted by two units, one that defines an object and another that defines an action) and is rewarded for the ability to exhibit the corresponding behaviour (i.e., executing the action on the target object). During training, an agent experiences only a subset of all possible object/action instructions. The goal of this work is to design neural mechanisms that allow the agent to access and execute both the experienced and the non experienced linguistic instructions, through the development of a compositional semantics that underpins linguistic and behavioural skills required by the task. This study has been strongly inspired by the work illustrated in [9] in which the authors trained a wheeled robot to interact with three coloured objects (located on the left, frontal, and right side of the agent) through three actions (indicating, hitting, and pushing). Also the idea of studying semantic combinatoriality through the co-development of linguistic and behavioural skills has been strongly inspired by the above seminal work in which the authors demonstrated how the linguistic and behavioural skills developed by the agents can be bounded together in order to allow the agent to react to a new linguistic instructions not experienced during training. Yet, we look at the problem with different methodological tools to provide further alternatives to those issues that we perceive as current limitations of the work described in [9]. In particular, in [9], the agent is controlled by two separated modules (one dedicated to perception and action, the other to linguistic comprehension) trained through a learning by demonstration process in which the sequence of sensory-motor states experienced while the experimenter drives the agent actuators during a demonstration session are used as teaching input for a supervised learning algorithm. Moreover, in [9] the sensory-motor module is trained to execute all the possible behaviours, even those associated to the linguistic instructions used to test the agent's generalisation capabilities. Contrary to [9], we propose to study the emergence of situated semantics in single non modularised artificial neural networks trained through a trial and error process (based on an evolutionary algorithm) in which the agents are rewarded on the basis of their ability to execute the linguistic instructions being free to determine how to execute such instructions. In our model, behavioural and linguistic competences co-evolve in a single neural structure in which the semantics is fully grounded on the sensory-motor capabilities of the agents and fully integrated with the neural mechanisms that underpin the agent's behavioural repertoire. Moreover, the agents are evolved to execute only the behaviours corresponding to the linguistics instructions experienced during training. Therefore, the capability of the agents to generalise concerns both the capability to access not experienced linguistic instructions as well as the capability to generate not experienced behaviours.

At the end of the training process successful agents display an ability to translate the linguistic instructions experienced during training into the corresponding situated behaviours. By analysing how successful agents react to specific combination of object/action instructions not experienced during training, we observed that some of the agents display an ability to spontaneously produce the appropriate behaviours, despite these behaviours have never been produced



	MOVE Act^M_o					
	Object			Action		
	I_4	I_5	I_6	I_7	I_8	I_9
Blue	0	0	1	1	0	0
Green	0	1	0	1	0	0
Red	1	0	0	1	0	0

	TOUCH Act^T_o					
	Object			Action		
	I_4	I_5	I_6	I_7	I_8	I_9
Blue	0	0	1	0	1	0
Green	0	1	0	0	1	0
Red	1	0	0	0	1	0

	INDICATE Act^I_o					
	Object			Action		
	I_4	I_5	I_6	I_7	I_8	I_9
Blue	0	0	1	0	0	1
Green	0	1	0	0	0	1
Red	1	0	0	0	0	1

(c)

Fig. 1. (a) The agent structure and its world. The vision system of the agent is drawn only with respect to the arm initialised on the right initialisation area. (b) The structure of neural network. Continuous line arrows indicate the efferent connections for the first neuron of each layer. Underneath the input layer, it is shown the correspondences between sensors/linguistic instructions, the notation used in equation 1a to refer to them, and the sensory neurons. (c) The linguistic instructions. In grey the instructions not experienced during training. INDICATE is considered only in Exp. A.

or rewarded before during training. Post-evaluation analyses on the behaviour of successful agents suggest that their capability to access unlearnt instructions and to generate the corresponding unlearnt behaviour partially results from the emergence of temporal (rather than topological as in [9]) structures of the semantic space. Finally, we observed that the development of systematised knowledge underpinned by a compositional semantic system is facilitated by evolutionary circumstances in which the agents are explicitly required to display elementary behavioural skills that can be recruited for the generation of more complex behaviours.

2 Methods

The task and the agent

Each agent lives in a two-dimensional world and is comprised of an arm with two segments referred to as S^1 (100 cm) and S^2 (50 cm), and two degrees of freedom (DOF). Each DOF is comprised of a rotational joint which acts as the fulcrum and an actuator. One actuator causes S^1 to rotate clockwise or anticlockwise around point O, with the movement restricted within the right ($-30°$) and the left ($210°$) bound. The other actuator causes S^2 to rotate within the range $[-90°, 90°]$ with respect to S^1. Friction and momentum are not considered (see Fig. 1a). In the environment there are three rounded objects of different colours (i.e., a blue, a green, and a red object). The objects are placed at 150 cm from point O with their centre placed anywhere on the cord delimiting their corresponding Init. sector (see Fig. 1a). The objects do not move unless pushed by the arm. The agent is equipped with a linear camera with a receptive field of 30°, divided in three sectors, each of which has three binary sensors (C_i^B for blue, C_i^G for green, and C_i^R for red, with $i \in [1, 2, 3]$ sectors). Each sensor returns 1 if the blue/green/red object falls with the corresponding sector. The camera and S^1 move together. The experimental set up is built in a way that at each time step there can be only one object in the camera view. If no coloured object is detected, the readings of the sensors are set to 0. The agent is also equipped with right and left bound binary sensors which activate (i.e., their reading is set to 1) whenever S^1 reaches the right or the left bound, respectively. Finally, three binary touch sensors (i.e., T^r, T^f, T^l) are placed on the right, front, and left side of S^2. Collisions between the agent and an object are handled by a simple model in which whenever S^2 pushes the object the relative contact points remain fixed.

In a first series of simulations (referred to as Exp. A), agents are trained to execute the following three actions: TOUCH (Act_o^T), MOVE (Act_o^M), and INDICATE (Act_o^I), where o is the object on which the action is executed, and can be either the *blue*, the *green* or the *red* object (see Fig. 1c). TOUCH requires an agent to remain in contact with the target object with the right side of S^2 (that is, by activating the touch sensor T^r) for an uninterrupted interval of 100 time steps. During this interval, S^1 does not have to rotate. MOVE requires an agent to rotate S^1 more than 35° while S^2 is touching the object with its right side. The rotation of S^1 while S^2 is touching the object determines the movement of the object. INDICATE requires an agent to rotate S^1 until the angular distance between S^1 and the object is less than 30°. INDICATE is correctly executed only if S^1 remains at less than 30° from the target object for more than 100 time steps. During the execution of INDICATE, an agent must not collide with any object. During the execution of TOUCH and MOVE, an agent must not collide with the non target objects (i.e., the objects not mentioned in the current linguistic instruction). In a second series of simulations (referred to as Exp. B), agents are trained to execute only the action TOUCH (Act_o^T), and MOVE (Act_o^M).

The agent controller and the evolutionary algorithm

The agent controller is composed of a continuous time recurrent neural network (CTRNN) of 20 sensor neurons, 3 inter-neurons and 4 motor neurons [10]. At each time step sensor neurons are activated using an input vector I_i with $i \in [1, .., 20]$ corresponding to the sensors readings (see Fig. 1b).

The inter-neuron network is fully connected. Additionally, each inter-neuron receives one incoming synapse from each sensory neuron. Each motor neuron receives one incoming synapse from each inter-neuron. There are no direct connections between sensory and motor neurons. The states of the motor neurons are used to control the movement of S^1 and S^2 as explained later. The states of the neurons are updated using the following equations:

$$\frac{\Delta y}{\Delta T} = \begin{cases} \Big(-y_i + gI_i\Big)\dfrac{1}{\Delta T}; & \text{for } i \in \{1,..,20\}; \quad \text{(1a)} \\ \Big(-y_i + \displaystyle\sum_{j=1}^{23} \omega_{ji}\sigma(y_j + \beta_j)\Big)\dfrac{1}{\tau_i}; & \text{for } i \in \{21,22,23\}; \quad \text{(1b)} \\ \Big(-y_i + \displaystyle\sum_{j=21}^{23} \omega_{ji}\sigma(y_j + \beta_j)\Big)\dfrac{1}{\Delta T}; & \text{for } i \in \{24,..,27\}; \quad \text{(1c)} \end{cases}$$

with $\sigma(x) = (1+e^{-x})^{-1}$. In these equations, using terms derived from an analogy with real neurons, y_i represents the cell potential, τ_i the decay constant, g is a gain factor, I_i the intensity of the perturbation on sensory neuron i, ω_{ji} the strength of the synaptic connection from neuron j to neuron i, β_j the bias term, $\sigma(y_j + \beta_j)$ the firing rate (hereafter, f_i). All sensory neurons share the same bias (β^I), and the same holds for all motor neurons (β^O). τ_i and β_i with $i \in \{21, 22, 23\}$, β^I, β^O, all the network connection weights ω_{ij}, and g are genetically specified networks' parameters. At each time step the angular movement of S^1 is $2.9H(f_{24} - 0.5)sgn(0.5 - f_{25})$ degrees and of S^2 is $2.9H(f_{26} - 0.5)sgn(0.5 - f_{27})$ degrees, where H is the Heaviside step function and sgn is the sign function.

A generational genetic algorithm is employed to set the parameters of the networks [11]. The population contains 100 genotypes. Generations following the first one are produced by a combination of selection with elitism, recombination and mutation. For each new generation, the five highest scoring individuals from the previous generation are retained unchanged. The remainder of the new population is generated by fitness-proportional selection from the 70 best individuals of the old population. Each genotype is a vector comprising 90 real values. Each gene is chosen uniformly random from the range $[0, 1]$. Cell potentials are set to 0 when the network is initialised or reset, and circuits are integrated using the forward Euler method with an integration time step $\Delta T = 0.05$.

The fitness function

During evolution, each genotype is translated into an arm controller and evaluated more than once for different object-action pairs and different starting positions. In Exp. A (i.e., with INDICATE), agents are evaluated 14 times initialised in the left and 14 times in the right initialisation area, for a total of 28 trials. For each initialisation area, an agent experiences 2 times all the linguistic instructions with the exception of Act^M_{blue} and Act^T_{green}. These two instructions are never experienced during the training phase. In Exp. B (i.e., without INDICATE), agents are evaluated 8 times initialised in the left and 8 times in the right initialisation area, for a total of 16 trials. 4 out of 6 linguistic instructions are experienced during the evolution process, while 2 are not (as before, the instructions which are not experienced are Act^{M}_{blue} and Act^{T}_{green}). In both Exp. A and Exp. B, at the beginning of each trial, the agent is randomly initialised in one of the two initialisation area, and the state of the neural controller is reset. A trial lasts 12 simulated seconds ($T = 250$ time steps). A trial is terminated earlier in case the arm collides with a non target object.

In each trial k, an agent is rewarded by an evaluation function which seeks to assess its ability to execute the desired action on the target object. The final fitness FF attributed to an agent is the sum of two fitness components F^1_k and F^2_k. F^1_k rewards the agent for reducing the angular distance between S^1 and the target object. F^2_k rewards the agent for performing the required action on the target object. F^1_k and F^2_k are computed as follows:

$$F^1_k = \max\left(0, \frac{d^i - d^f}{d^i} \cdot P^1_k, \mathbb{1}_{d^f < 4.6^\circ}\right) \tag{2}$$

where d^i and d^f are respectively the initial (i.e., at $t = 0$) and final (i.e., at the end of the trail k) angular distances between S^1 and the target object and $\mathbb{1}_{d^f<4.6^\circ}$ is 1 if $d^f < 4.6^\circ$, 0 otherwise. P^1_k is the penalty factor, which is set to 0.6 if the agent collides with a non target object, to 1.0 otherwise. The angle between S^1 and the target object o can be measured *clockwise* (α^{clock}_o) or *anticlockwise* (α^{anti}_o). In equation 2, d^i and d^f are the minimum between the clockwise and anticlockwise distance, that is $d = \min\left(\alpha^{clock}_T, \alpha^{anti}_T\right)$.

$$F^2_k = \begin{cases} \dfrac{\textit{steps-on-target}}{\textit{max-steps-on-target}} \cdot P^2_k & \text{for TOUCH or INDICATE} \quad (3a) \\ \dfrac{\Delta\theta}{\textit{max-angular-offset}} \cdot P^2_k & \text{for MOVE} \quad (3b) \end{cases}$$

where *max-steps-on-target* $= 100$, $P^2_k = 0$ if $F^1_k < 1$ otherwise $P^2_k = 1$, and *max-angular-offset* $= 34.4^\circ$. For the action INDICATE, *steps-on-target* refers to the number of time steps during which $F^1_k = 1$, and S^2 does not touch the target object. For the action TOUCH, *steps-on-target* refers to the number of time steps during which $F^1_k = 1$, S^2 touches the target object by activating the touch sensor T^r, and S^1 does not change its angular position. $\Delta\theta$ is the angular displacement of the orientation of S^1 recorded while $F^1_k = 1$, and S^2 is touching the target object by activating the touch sensor T^r. A trial is terminated earlier if *steps-on-target* = *max-steps-on-target* during the execution of INDICATE or TOUCH and when $\Delta\theta$ = *max-angular-offset* during the execution of MOVE.

3 Results

For both Exp. A and Exp. B, we run for 10000 generations ten evolutionary simulations, each using a different random initialisation. Recall that our objective is to generate agents that are capable of successfully performing all the possible

Table 1. Result of post-evaluation test performed on the best agents of each generation for each run and for Exp. A and Exp. B. The table shows the number of successful agents on linguistic instructions experienced during evolution, and the percentage of successful agents on linguistic instructions not experienced during evolution indicated by the corresponding row (see text for details).

run		n. 1	n. 2	n. 3	n. 4	n. 5	n. 6	n. 7	n. 8	n. 9	n. 10
Exp. A											
Num. Suc. Agents		8634	0	7182	0	5491	3466	8812	8312	4627	8632
(%)	Act^M_{blue}	30.87	0.00	17.96	0.00	0.00	57.73	29.43	27.96	12.19	3.56
	Act^T_{green}	17.88	0.00	0.56	0.00	2.77	1.13	16.00	21.19	3.41	1.00
	Act^M_{blue} and Act^T_{green}	9.07	0.00	0.61	0.00	0.00	1.59	6.97	15.56	0.35	0.00
Exp. B											
Num. Suc. Agents		6044	6011	8689	8893	0	8385	9060	7620	9151	8304
(%)	Act^M_{blue}	20.43	14.59	11.67	19.98	0.00	0.01	1.10	16.18	3.05	7.70
	Act^T_{green}	0.00	0.32	1.63	2.11	0.00	10.10	1.62	0.59	1.22	0.87
	Act^M_{blue} and Act^T_{green}	0.00	0.00	0.44	0.16	0.00	0.00	0.00	0.21	0.00	0.00

behaviours corresponding to the execution of all the possible linguistic instructions by undertaking a training focused only on a subset of them. We run two different series of simulations (i.e., Exp. A and Exp. B) to see whether the training on a more elementary action (i.e., INDICATE) bears upon the development of functionally compositional neural structures.

The best agents of each generation in both experimental conditions have been post-evaluated by running sets of 80 trials for each linguistic instruction. Agents of Exp. B are not tested on linguistic instructions that require action INDICATE. In half of the trials the agents are randomly initialised in the right and half of the trials in the left initialisation area (see Fig 1a). We considered successful at the post-evaluation tests the agents that managed to obtain a success rate higher that 80% in performing the behaviours corresponding to the execution of the linguistic instructions experienced during evolution. Successful agents have been further classified in i) *non compositional* agents, referring to those successful agents that proved to be less than 80% successful at performing the behaviour corresponding to the execution of both the not experienced instructions, Act^M_{blue} and Act^T_{green}; ii) *partially compositional* agents referring to those successful agents that proved to be more than 80% successful at performing the behaviour corresponding to the execution of only one of the two non experienced instructions, Act^M_{blue} or Act^T_{green}; iii) *fully compositional* agents referring to those successful agents that proved to be more than 80% successful at performing the behaviour corresponding to the execution of both the not experienced instructions, Act^M_{blue} and Act^T_{green}. Results of post-evaluation tests are shown in Table 1.

All the runs, with the exception of run n. 2 and n. 4 in Exp. A, and run n. 5 in Exp. B, generated plenty of successful agents. For what concerns com-

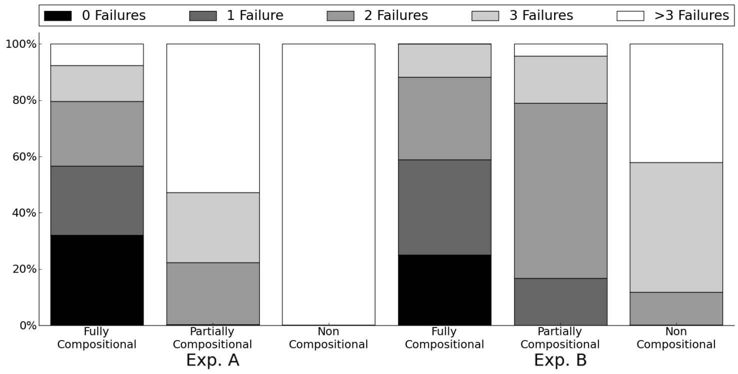


Fig. 2. Percentage of *fully compositional*, *partially compositional* and *non compositional* agents in the two experimental conditions grouped by the number of failure.

positionality, the results can be summarised in few relevant points. First, *fully compositional* agents are a very small percentage of the successful agents, in Exp A, and they are almost absent in Exp. B. Moreover, in those run that generated them, *fully compositional* agents keep on appearing and disappearing during evolution, while successful agents once generated they are almost never lost (data not shown). These data suggest that compositionality is not automatically associated with, and is not a prerequisite for developing the capability of successfully performing the evolutionary task. Second, in both Exp. A and Exp. B,*partially compositional* agents are slightly more frequent than fully compositional agents. Moreover, *partially compositional* agents capable of performing Act^M_{blue} are more frequent than *partially compositional* agents capable of performing Act^T_{green}. Third, although successful agents are slightly less likely to be generated in Exp. A than Exp. B, *fully compositional* or *partially compositional* agents are definitely more frequent in Exp. A than in Exp B. This suggests that the training on the more elementary action INDICATE seems to facilitate the development of behavioural and linguistic compositionality.

Having ascertained that some of the successful agents are also *partially* or *fully compositional*, we try to understand more about the mechanisms underpinning compositionality. Looking at the behaviour of all types of *compositional* agents, we noticed that they first move S^1 keeping S^2 bent in order to point to the target object (as required for the INDICATE instruction). After that, if TOUCH or MOVE is required, they rotate S^2 and eventually S^1 again depending on the current linguistic instruction. If INDICATE is required, they keep S^1 pointing to the object, and S^2 fully bent as at start. A very parsimonious hypothesis on how *compositional* agents generate these behavioural patterns is based on the capability to "parse" the linguistic instruction and to "pay attention" to its parts in a sequential order. According to our temporal sequencing hypothesis, compositionality may result from the fact that at the beginning of a trial, when the agents have to approach the target object, only the part of the instruction referring to the object bears on its behaviour. When an agent is ready to execute the action on the target object, then only the part of the instruction referring to the action bears on the agent behaviour. In other words, compositionality may be underpinned by a systematised knowledge of the task obtained by paying attention to different parts of the linguistic instruction at different times of a trial. Linguistic instructions, including those not experienced during training, would be "decomposed" in already experienced elementary units which trigger known (i.e., already experienced) elementary behaviours in a specific temporal sequence (i.e., first the movement on the target object, then the execution of the desired action).

To test the temporal sequencing hypothesis, we run a further series of post-evaluation tests on successful agents of both Exp. A and Exp. B. In these tests, the linguistic command referring to the action is changed during the agents' life time as soon as the agents have completed the movement toward the target object (i.e., when $d^f < 0.08$, see Sec. 2). According to the temporal sequencing hypothesis, *compositional* agents should pay attention to the part of the linguistic instruction referring to the action only after having reached the target object. Therefore, they should correctly execute the second-given action, while ignoring the first-given one. The performance of *non compositional* agents should result severely disrupted by this type of unexpected manipulation of the linguistic instruction. The agents undergo sets of 80 trials for each possible transition from a first-given action to a second-given action different from the first one, and for each object. In half of the trials the agents are randomly initialised in the right and half of the trials in the left initialisation area. There are 18 possible transitions in Exp. A and 6 in Exp. B. The performance of an agent on each specific transition is considered a failure if the agent fails to execute the second-given action in more than 64 out of 80 trials. The results shown in Fig. 2 indicate that only some of the *fully compositional* agents are able to perform all transitions without any failure. These agents appear to have acquired a systematised knowledge of the task in accordance with what suggested by the temporal sequencing hypothesis. The higher the number of failure, the less structured the knowledge of the task with a higher number of linguistic instructions learnt by rote and represented as "atomic" operations in a semantics space progressively less compositional. Note that it is possible to be a *compositional* agent and having few linguistic instructions learnt by rote. This is probably the case of *fully compositional* agents that make several failure on specific transitions (remember that we do not enforce by any means compositionality). Note also that Exp. A and Exp. B generate similar results. This may imply that *fully compositional* agents exploit the same mechanisms to achieve compositionality in spite of the fact that in Exp. B the evolutionary conditions seem not to facilitate their evolution.

4 Conclusions

The results of this study shows that dynamical neural networks designed by artificial evolution can provide the required mechanism to develop a compositional semantic neural structures which allow autonomous agents to access linguistic instructions not experienced during training and to execute the corresponding behaviours also non experienced during training. Although we haven't carried out yet any analysis on the neural mechanisms, we run some behavioural tests which showed that evolved compositional semantic systems seem to be underpinned by temporal structures. That is, *fully compositional* agents possess the required mechanisms to "parse" different part of the instruction and to execute different sub-behaviours at different time of their life span. Evolutionary conditions in which the agents are explicitly required to execute more elementary behaviour than those on which their compositional skills are evaluated seem to facilitate the emergence of *fully compositional* agents. Leaving the agents free to determine how to achieve the goals associated to each linguistic instruction allowed the agents to organise their behavioural skills in ways that facilitate the development of compositionality thus enabling the possibility to display a generalisation ability at the level of behaviours (i.e., the ability to spontaneously produce new behaviours that have not been displayed or rewarded before). In future research we plan to investigate the characteristics that favour the emergence of compositional solutions (that ensure behavioural generalisation) and/or that reduce the chance to converge on non-compositional solutions and the possibility to scale the model with respect to the number and the complexity of the linguistic/behavioural repertoire.

References

1. Cangelosi, A., Parisi, D., eds.: Simulating the evolution of language. Springer Verlag, New York (2002)
2. Roy, D.: Grounding words in perception and action: computational insights. Trends in Cognitive Sciences **9**(8) (2005) 389–396
3. Steels, L.: Experiments on the emergence of human communication. Trends in Cognitive Sciences **10**(8) (2008) 347–349
4. Nolfi, S., Mirolli, M., eds.: Evolution of Communication and Language in Embodied Agents. Springer Verlag, Berlin (2010)
5. Glenberg, A., Kaschak, M.: Grounding language in action. Psychonomic Bulletin & Review **9** (20002) 558–565
6. Rizzolatti, G., Arbib, M.: Language within our grasp. Trends in Neuroscience **21** (1998) 188–194
7. Pulvermuller, F.: The Neuroscience of Language. On Brain Circuits of Words and Serial Order. Cambridge University Press (2003)
8. Cappa, S., Perani, D.: The neural correlates of noun and verb processing. Journal of Neurolinguistics **16**(2–3) (2003) 183–189
9. Sugita, Y., Tani, J.: Learning semantic combinatoriality from the intercation between linguistic and behvioral processes. Adaptive Behavior **13**(1) (2005) 33–52
10. Beer, R.D., Gallagher, J.C.: Evolving dynamic neural networks for adaptive behavior. Adaptive Behavior **1**(1) (1992) 91–122
11. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA (1989)