

Self-Organising Sync in a Robotic Swarm.

A Dynamical System View

Vito Trianni and Stefano Nolfi

Laboratory of Autonomous Robotics and Artificial Life (LARAL),

Institute of Cognitive Sciences and Technologies (ISTC)

National Research Council (CNR), Rome, Italy.

e-mail: {vito.trianni,stefano.nolfi}@istc.cnr.it

Abstract

Self-organised synchronisation is a common phenomenon observed in many natural and artificial systems: simple coupling rules at the level of the individual components of the system result in an overall coherent behaviour. Owing to these properties, synchronisation appears particularly interesting for swarm robotics systems, as it allows for robust temporal coordination of the group while minimising the complexity of the individual controllers. The goal of the experiments presented in this paper is the study of self-organising synchronisation for robots that present an individual periodic behaviour. In order to design the robot controllers, we make use of artificial evolution, which proves to be capable of synthesising minimal synchronisation strategies based on the dynamical coupling between robots and environment. The obtained results are analysed under a dynamical system perspective, which allows us to uncover the evolved mechanisms and to predict the scalability properties of the self-organising synchronisation with respect to varying group size.

Keywords: Self-Organisation, Synchronisation, Swarm Robotics, Dynamical Systems

1 Introduction

Synchrony is a pervasive phenomenon: examples of synchronous behaviours can be found in the inanimate world as well as among living organisms [1, 2]. Synchrony may spontaneously emerge from weak interactions among coupled systems: the synchronisation of pendulums reported by Huygens is probably one of the first documented examples [3]. In biological systems too, synchrony is often observable. For instance, the heart pacemaker cells synchronise to achieve a robust beat, resulting in a system resilient to failures of individual cells [4, 5]. Synchronisation among neurons leads to the formation of assemblies of coherent activity that are considered to be at the basis of cognitive processes such as binding—i.e., the integration of information from different sensory perceptions of the same phenomenon—, selective attention, learning and memory [6–12]. Similar mechanisms are at the base of the synchronous signalling behaviour observed in various animal species [13]. “Chorusing” is a term commonly used to refer to the coordinated emission of acoustic communication signals by large groups of animals. To cite a few, chorusing has been observed in frogs [14], crickets [15] and spiders [16]. It has also been argued that synchronous chorusing in hominids may have played a fundamental role in the evolution of language and music [17]. Synchronous displays that do not involve acoustic signals have been extensively studied in fireflies, which emit coordinated light pulses [18], and in crabs, which wave their claws in synchrony [19]. Other synchronisation phenomena may have a catastrophic outcome: from epileptic seizures, the Parkinson’s disease or schizophrenia, which are all caused by excessive synchrony in some areas of the human brain [20–22], to the side-swaying of the London Millennium Bridge, which was caused by spontaneous synchronisation of the walking pace of the many people traversing it on the opening day [23]. Much research has been dedicated also to the discovery of synchronisation from observable data in noisy or chaotic conditions, where only the phase locking is relevant while the amplitudes have no restrictions [24, 25]. For this purpose, analytical tools have

been introduced based on the definition of the “instant frequency” of the given signals [26, 27].

How can all these systems—so much diverse from each other—self-organise to achieve synchrony? This question aroused the research of an answer for many years, until the appropriate analytical methods were developed [28, 29]. All the above synchronisation phenomena can be modelled as systems of multiple coupled oscillators. Consider for example the synchronous flashing of fireflies: thousands of insects emit light pulses in unison, perfectly synchronising their individual rhythm. In this case, fireflies can be modelled as a population of pulse-coupled oscillators with equal or very similar frequencies. These oscillators can influence each other by emitting a pulse that shifts or resets their oscillation phase. The numerous interactions among the individual oscillator-fireflies are sufficient to explain the synchronisation of the whole population (for more detail, see [4, 5, 18]). Despite the clear understanding of the mechanism, the functionality of synchronisation or, in the particular case of animal behaviour, its adaptive significance is not always clear. With respect to chorusing behaviours, the most convincing hypothesis is that synchrony is an epiphenomenon of the competition between males to attract females, and results from the attempt of each individual to anticipate the signal of its neighbours [30].

The synchronisation behaviours observed in Nature can be a powerful source of inspiration for the design of robotic systems. From manipulators [31] to hexapod robots [32], synchronisation is an important mean to achieve coordination. This holds particularly for swarm robotics systems [33], in which the emergence of coherent group behaviours from simple individual rules is emphasised. Chorusing was the metaphor for the coordination algorithm used in a collective robotics experiment, in order to regulate the group size and let the robots coordinately move towards a target location [34, 35]. Other works take inspiration from the self-organising behaviour of fireflies: a specialised neural module was designed for the synchronisation of the foraging/homing activities in a robot group in order to maximise the overall performance [36]. The same mecha-

nism was also applied to a cleaning task to be performed by a swarm of micro robots [37]. Finally, similar synchronising behaviours could be synthesised by artificial evolution as adaptive mechanisms to reduce the interference among communicating robots [38].

The goal of the experiments presented in this paper is the study of self-organising synchronisation in a group of robots based on minimal behavioural and communication strategies. Similar to the studies presented above, we follow the basic idea that if an individual displays a periodic behaviour, it can synchronise with other (nearly) identical individuals by temporarily modifying its behaviour in order to reduce the phase difference with the rest of the group. In other studies, synchronisation is based on the entrainment of the individual internal dynamics through some form of communication. In this paper, instead, we do not postulate the need of internal dynamics. Rather, the period and the phase of the individual behaviour are defined by the sensory-motor coordination of the robot [39], that is, by the dynamical interactions with the environment that result from the robot embodiment. We show that such dynamical interactions can be exploited for synchronisation, allowing to keep a minimal complexity of both the behavioural and the communication level. Now, the main problem is defining a robot controller able to exploit the dynamical agent-environment interactions. By relying on a simple kinematic simulator of our robots, we use artificial evolution to search the space of the possible behavioural and communication strategies for the synchronisation problem [40,41]. In particular, we avoid to explicitly reward the use of communication, in order to leave evolution free to explore the space of the possible solutions that lead to synchronous behaviour and to allow the evolving robots to co-adapt their behavioural and communication skills. The obtained results are analysed under a self-organising perspective, evaluating the scalability to large groups of robots. Moreover, we investigate the scalability of the synchronisation mechanism *per se* in order to evaluate the efficiency of the evolved strategy when not constrained by the physical interactions among the robots. Additionally, we test

the behaviours evolved in simulation with physical robots, therefore providing a proof-of-concept about the viability of the proposed methodology for robot controller design.

The main contribution of this paper consists in the analysis of the evolved behaviours, which employs a dynamical system approach [42]. Dynamical systems theory is recently acquiring more and more attention in cognitive sciences [43–45] as it can give explanations of cognitive phenomena while they unfold over time. Concepts like “attractor” and “bifurcation” start to be commonly used, and dynamical models are developed—just to name a few—to give new answers to classic psychology debates such as the A-not-B error in infant reaching [46], or to account for intrinsically dynamical processes such as inter-limb coordination [47,48]. In this paper, we introduce a dynamical system model of the evolved behaviours, in order to uncover the mechanisms that artificial evolution synthesised to maximise the user-defined utility function. Moreover, we show how the developed model can be used to predict the ability of the evolved behaviour to efficiently scale with the group size. We believe that such predictions are of fundamental importance to quickly select or discard obtained solutions without performing a time-demanding scalability analysis, as well as to engineer swarm robotics systems that present the desired properties.

This paper is organised as follows. In Section 2, we present the experimental setup devised to evolve the self-organising synchronisation behaviours in a simple simulation environment. Section 3 briefly summarises the results obtained from the evolutionary machinery. In Section 4, we provide an analysis of the evolved behaviours from a dynamical system perspective. In particular, Section 4.1 introduces a mathematical model of the behaviours evolved in simulation, which includes some simplifications (e.g., neglecting noise) necessary for the dynamical system analysis. As we shall discuss, such simplifications do not influence the relevant aspects of the individual behaviour and of the synchronisation dynamics, which are described in Sections 4.2 and 4.3. Section 5 is dedicated to the scalability properties of the evolved behaviours and

of the synchronisation mechanism. These analyses are performed exploiting the simulation environment used for evolutionary optimisation. Additionally, in Section 5.3 we show how the mathematical model can be exploited to predict the scalability of the evolved controllers on the basis of the characteristics of the individual behaviour. Section 6 discusses the results obtained by testing the controllers evolved in simulation with physical robots. Finally, Section 7 concludes the paper with some discussions about the proposed approach and the obtained results.

2 Evolution of Self-Organising Synchronisation

In this section, we present the experimental setup defined for the evolution of synchronisation behaviours. The scenario we propose is simple and idealised. Nevertheless, it contains all the ingredients necessary to study self-organised synchronisation in a swarm of robots. The task requires that each robot in the group displays a simple periodic behaviour, which should be entrained with the periodic behaviour of the other robots in the arena. The individual periodic behaviour consists in oscillations along the y direction of the rectangular arena (see Figure 1). Oscillations are possible through the exploitation of a symmetric gradient in shades of grey painted on the ground. On the other hand, synchronisation of the robots' movements can be achieved by exploiting communication.

Robot controllers are evolved in a simulated environment. The simulated arena is a rectangle of 6×3 m completely surrounded by walls. We set an xy reference frame as shown in Figure 1. The ground is painted in white for $|y| < 0.2$ m, and linearly changes to black until $|y| = 1$ m. For larger distances, the arena is painted in black. Robots should oscillate on the painted gradient without moving over the black area. As a consequence, robots can make oscillations with a maximum amplitude of 2 m. Given the symmetry of the painted gradient, synchronised movements correspond to both in-phase or anti-phase oscillations. As an example, in Figure 2 we show the y position of three robots

that perform synchronous oscillations. While the first two robots display in-phase oscillations, the third robot displays anti-phase oscillations with respect to the others (see Section 2.3 for more detail). In the following, we give further details about the experimental setup by describing the robotic platform used (see Section 2.1), the controller and the evolutionary setup (see Section 2.2) and finally the fitness function used (see Section 2.3).

2.1 The Robots

The robots used in these experiments are *s-bots* (see Figure 3), which are small autonomous robots with the ability to self-assemble [49, 50].¹ The evolutionary experiments presented in this paper are performed in simulation, using a simple kinematic model of the *s-bots*, and the results are afterwards validated on the physical platform.

An *s-bot* weighs 700 g and its main body has a diameter of about 12 cm. Its design is innovative with respect to both sensors and actuators. The traction system is composed of both tracks and wheels, each track-wheel pair of the same side being controlled by a single motor. This combination of tracks and wheels provides the *s-bot* with a differential drive motion, which is labelled *Differential Treels*[®] *Drive*. The treels are connected to the chassis, which contains the batteries. The main body is a cylindrical turret mounted on the chassis by means of a motorised joint that allows the relative rotation of the two parts (see Figure 3). An *s-bot* is provided with many sensory systems useful for the perception of the surrounding environment or for proprioception. Infrared proximity sensors are distributed around the rotating turret and can be used for detection of obstacles and other *s-bots*. Four proximity sensors placed under the chassis—referred to as *ground sensors*—can be used for perceiving the grey level of the ground, the presence of holes or the terrain’s roughness (see Figure 3). Additionally, an *s-bot* is provided with light sensors, temperature/humidity sensors, a 3-axes

¹The assembling capability of the *s-bots* is not the focus of these experiments. For more detail on self-assembling *s-bots*, see [51].

accelerometer and incremental encoders on each degree of freedom. Each robot is also equipped with sensors and devices to detect and communicate with other *s-bots*, such as an omni-directional camera, coloured LEDs around the *s-bots*' turret, microphones and loudspeakers (see Figure 3).

In the experiments described in this paper, we only use the infrared and ground sensors for perceiving the environmental features, the loudspeaker and the microphones for sound signalling, and the two motors controlling the turrets. In particular, the loudspeaker and the microphones are used to implement a global, binary communication system. The loudspeaker can be used to emit a binary signal with a fixed frequency $\mathcal{F} = 2500 \text{ Hz}$ and an intensity high enough to be perceived from anywhere in the experimental arena. On the receiver side, at every control cycle the recordings from the microphones are processed by the on-board CPU to extract the frequency spectrum. If the intensity perceived in a small interval around \mathcal{F} is above a certain threshold, the binary sound sensor is set to 1. To summarise, each *s-bot* can produce a continuous tone with fixed frequency and intensity. When a tone is emitted, it is perceived by every robot in the arena, including the signalling *s-bot*. The tone is perceived in a binary way, that is, either there is at least one *s-bot* signalling in the arena, or there is none.

2.2 The Controller and the Evolutionary Algorithm

Artificial evolution is used to set the connection weights and the bias terms of simple neural controllers with fixed architecture. The controller of each *s-bot* is a fully connected, feed forward neural network—a perceptron network. The neural network has 11 sensory neurons directly connected to 3 motor neurons. The sensory neurons are simple relay units and the output neurons are sigmoid units whose activation is computed as follows:

$$O_j = \sigma \left(\sum_i w_{ij} I_i + \beta_j \right), \quad \sigma(z) = \frac{1}{1 + e^{-z}}, \quad (1)$$

where I_i is the activation of the i^{th} input unit, β_j is the bias term, O_j is the activation of the j^{th} output unit, w_{ij} is the weight of the connection between the input neuron i and the output neuron j , and $\sigma(z)$ is the sigmoid function.

Six sensory neurons— I_1 to I_6 —receive input from a subset of the infrared proximity sensors evenly distributed around the *s-bot*'s turret. Four sensory neurons— I_7 to I_{10} —are dedicated to the readings of the four ground sensors. The state of all infrared and ground sensors is linearly scaled to the range $[0.0, 1.0]$. A simulated uniform noise within 5% of the input range is also added. The last sensory neuron I_{11} receives a binary input corresponding to the perception of sound signals. The activation states of the first two motor neurons— O_1 and O_2 —is scaled onto the range $[-\omega_M, +\omega_M]$, where ω_M is the maximum angular speed of the wheels ($\omega_M \approx 4.5 \text{ s}^{-1}$). The third motor neuron controls the speaker in such a way that a sound signal is emitted whenever the activation state O_3 is greater than 0.5.

The evolutionary algorithm is based on a population of 100 genotypes, which are randomly generated. This population of genotypes encodes the connection weights of 100 neural controllers. Each connection weight is represented with a 8-bit binary code mapped onto a real number ranging in $[-10, +10]$. Subsequent generations are produced by a combination of selection with elitism and mutation. Recombination is not used. At each generation, the 4 best individuals—i.e., the *elite*—are retained in the subsequent generation. The remainder of the population is generated by mutation of the 20 best individuals. Each genotype reproduces at most 5 times by applying mutation with 3% probability of flipping a bit. The evolutionary process runs for 500 generations. Simple algorithms of this type—i.e., mutation only and binary encoding of neural network weights—are widely used in the evolutionary robotics domain (for a review, see [40, 41]).

2.3 The Fitness Computation

During evolution, a genotype is mapped into a control structure that is cloned and downloaded onto all the *s-bots* taking part in the experiment (i.e., we make use of a homogeneous group of *s-bots*). Each genotype is evaluated 10 times—i.e., for 10 trials. Each trial differs from the others in the initialisation of the random number generator, which influences the initial positions and orientations of the *s-bots* within the arena. Each trial lasts $T = 900$ simulation cycles, which correspond to 90 seconds of real time.

The fitness of a genotype is the average performance computed over the 10 trials in which the corresponding neural controller is tested. During a single trial, the behaviour produced by the evolved controller is evaluated by a 2-component fitness function: $F = 0.5 \cdot F_{\mathcal{M}} + 0.5 \cdot F_{\mathcal{S}} \in [0, 1]$. The movement component $F_{\mathcal{M}}$ rewards robots that move along the y direction within the arena:

$$F_{\mathcal{M}} = \frac{1}{TR} \sum_{r=1}^R \sum_{t=1}^T \frac{|\Delta y(t, r)|}{\Delta Y}, \quad (2)$$

where $R = 3$ is the total number of robots in the group, $\Delta y(t, r)$ is the variation of the y position of *s-bot* r at cycle t , and ΔY is the maximum possible variation, which corresponds to the *s-bot* moving at maximum speed in a direction parallel to the y axis. This fitness component rewards fast motion along the y direction. The oscillatory behaviour evolves because the arena is surrounded by walls and by the black-painted area, so that oscillations during the whole trial are necessary to maximise $F_{\mathcal{M}}$.

The second fitness component $F_{\mathcal{S}}$ rewards synchrony among the robots. Synchrony among two *s-bots* can be evaluated as the cross-correlation coefficient between the sequences of movements parallel to the y axis performed during a trial. In order to encode the *s-bot* movements, we define the following function:

$$d(t, r) = y(t, r) \cdot \frac{\Delta y(t, r)}{\Delta Y}, \quad (3)$$

which depends on the y position of *s-bot* r at time t and on its displacement along the y direction. We chose to consider not only the absolute position but also the displacement in order to take into account the direction of oscillatory movements—e.g., away from or towards the x axis. The cross-correlation coefficient $\phi_{r_1 r_2}$ of two sequences $d(t, r_1)$ and $d(t, r_2)$ can be defined as:

$$\phi_{r_1 r_2} = \frac{\Phi_{r_1 r_2}}{\sqrt{\Phi_{r_1 r_1} \Phi_{r_2 r_2}}}, \quad \Phi_{r_1 r_2} = \frac{1}{T} \sum_{t=1}^T d(t, r_1) d(t, r_2). \quad (4)$$

The coefficient $\phi_{r_1 r_2}$ can take values in $[-1, 1]$, where a value of 1 indicates perfect synchrony and a value of -1 indicates perfect asynchrony. Notice that, given the symmetry of $d(t, r)$ with respect to movements away from or towards the x axis—see equation (3)—synchrony is rewarded also when robots perform the same movements at the same distance from the x axis, but on opposite sides of the x axis and in opposite directions. This is necessary due to the symmetric gradient in the arena, as shown in Figure 1, which results in identical perception of the gradient by the robots in the upper and lower part of the arena. Given equation (4), the synchrony component F_S is computed as the minimum among the cross-correlation coefficients of all possible pairs $\langle r_1, r_2 \rangle$ among the *s-bots*, bounded in $[0, 1]$:

$$F_S = \max\{0, \min_{r_1 \neq r_2} \phi_{r_1 r_2}\}. \quad (5)$$

In addition to the fitness computation described above, two indirect selective pressures are present. First of all, a trial is stopped when an *s-bot* moves over the black-painted area, and we assign to the trial a performance $F = 0$. In this way, robots are rewarded to exploit the information coming from the ground sensors to perform the individual oscillatory movements. Secondly, a trial is stopped when an *s-bot* collides with the walls or with another robot, and also in this case we set $F = 0$. In this way, robots are evolved to avoid collisions.

3 Results

We performed 20 evolutionary replications, each starting with a different population of randomly generated genotypes. Each replication produced a successful synchronisation behaviour, in which robots display oscillatory movements along the y direction and synchronise with each other according to the requirements of the fitness function. To assess the quality of the evolved behaviours, we select a single genotype per evolutionary replication to be chosen among the best individuals of the final generation. To do so, we evaluate the performance of the 20 best individuals of the final generation in 500 different trials, and we choose the individual with highest average fitness. In the remainder of the paper, we refer to the best controllers evolved in replication i as $c_i, i = 1, \dots, 20$. The performance of these controllers over the 500 post-evaluation trials, sorted according to decreasing median values, is shown in Figure 4. The obtained results show that in most replications the performance obtained is in average within the interval $[0.7, 0.9]$, which indicates that robots are able to maximise both the movement fitness component, F_M , and the synchronisation component, F_S . In order to assess the difference in performance among the controllers evolved in different evolutionary replications, we used the performance data recorded over 500 trials to perform a series of pairwise Wilcoxon tests among all possible controller couples. The results are plotted in Figure 4 as vertical lines spanning over the controller numbers having a performance that is not statistically different (at 99% confidence). So, for example, controllers c_{13} and c_{15} are not statistically different from the performance point of view. Similarly, controller c_1 has a performance equivalent to c_{18} and c_{19} , but it performs worse than controllers c_{10} and c_{14} . As can be seen in Figure 4, controller c_8 outperforms all other controllers. In the following, we give a detailed analysis of the behaviour produced by c_8 and by other controllers, in order to uncover the mechanisms that lie behind the evolved synchronisation behaviours.

4 Behavioural Analysis

A qualitative analysis of the obtained controllers reveals that the behaviours produced are quite similar to one another.² In general, it is possible to distinguish two phases in the evolved behaviours: an initial transitory phase during which robots achieve synchronisation, and a subsequent synchronised phase. The transitory phase may be characterised by physical interferences between robots due to collision avoidance, if robots are initialised close to each other. The collision avoidance behaviour performed in this condition eventually leads to a separation of the *s-bots* in the environment, so that further interferences to the individual oscillations are limited and synchronisation can be achieved. During the synchronous phase, collision avoidance is therefore less probable, but still possible due to the environmental noise, which may make robots deviate from their normal movements and approach other robots. Otherwise, this phase is characterised by stable synchronous oscillations of all *s-bots*, and small deviations from synchrony are immediately compensated.

In all replications, *s-bots* present periodic oscillations with varying amplitude on the *y* direction. Concerning the synchronisation mechanisms, it is possible to classify the evolved solutions into two main classes. The first class is characterised by a synchronisation mechanism that we refer to as the *modulation mechanism*: *s-bots* synchronise by modulating their oscillatory behaviour in response to a perceived communication signal coming from other robots. This class is composed of 15 controllers, including the best evolved controller c_8 . The second class includes the remaining 5 controllers, and is characterised by a synchronisation mechanism that we refer to as the *reset mechanism*: in response to the perceived communication signal, *s-bots* “reset” their oscillation by moving to a particular position over the painted gradient, waiting for the other robots to reach a similar position. We selected controller c_{10} to study the properties of the reset mechanism, since this controller has the highest performance within its

²Videos of the evolved behaviours can be found at <http://laral.istc.cnr.it/esm/trianni-nolfi-ieetec09>.

class. In Section 4.2, we give a quantitative analysis of the individual behaviour produced by controllers c_8 and c_{10} . Subsequently, Section 4.3 is dedicated to the synchronisation behaviour. In both cases, we make use of concepts borrowed from dynamical systems theory. To do so, we model our robotic system as a discrete-time dynamical system, which is the subject of the following section.

4.1 Dynamical System Modelling

We want to analyse the behaviour of a group of robots that synchronise their periodic oscillations. Our main interest is the understanding of both the individual behaviour—i.e., the periodic oscillation—and the synchronisation mechanism. Such understanding may be useful to predict some features of the evolved behaviour, e.g., the scalability discussed in Section 5.3. However, some simplifications are necessary for such a study. First of all, we neglect the collision avoidance behaviour among robots and between robots and the arena walls, as if robots were placed in an infinite arena, in which the chances to encounter another robot are null (*assumption I*). This seems a strong simplification, above all for what concerns the group behaviour, which may be deeply influenced by physical interactions and collision avoidance (in this respect, see Section 5.1). However, notice that to the extent of describing the individual oscillatory behaviour, collision avoidance with walls does not play a major role because robots are evolved to exploit the grey gradient for their movements. Similarly, the synchronisation mechanism does not rely on collision avoidance among robots, since evolution was performed with a rather low density of robots, so that physical interactions are not frequent enough to be exploited for synchronisation. Observations of the evolved behaviours confirm that collision avoidance can be neglected for our purposes. We will consider physical interactions again in the scalability analysis and when real world scenarios are considered (see Section 5 and Section 6). A second simplification is to ignore any form of noise, assuming that it is not exploited by the evolved behaviours (*assumption II*). Also in this case, preliminary behavioural analyses suggested that noise is not relevant for the production of

the individual oscillatory behaviour or for synchronisation. The third simplification concerns the dynamics of the individual robot: we neglect all second order dynamics such as acceleration and inertia (*assumption III*). The reason is twofold. On the one hand, the simulator used to evolve the robot controllers is kinematic, and already neglects second order dynamics. On the other hand, the maximum speed of the *s-bots* is rather low (about $0.112m/s$) while friction of the treels is rather high, so that an *s-bot* can accelerate to the maximum speed or completely stop in a single control cycle.

Consider a single robot placed in the experimental arena. On the basis of the simplifying assumptions described above, its behaviour can be completely characterised by the interaction with an idealised noise-free environment, mediated by the control rules encoded in the neural network controller. The neural controller is reactive and can be described as a function that associates the sensor inputs to the motor outputs:

$$\mathbf{O}(t) = \mathcal{N}(\mathbf{I}(t), \mathbf{c}), \quad (6)$$

where $\mathbf{I}(t)$ and $\mathbf{O}(t)$ represent the vector of inputs and outputs of the neural network, and \mathbf{c} is the vector of evolved parameters that characterise the neural controller. Notice that the analytical form of function \mathcal{N} is given in equation (1). The input vector is defined by the sensor readings of the robot. Here, we make use of assumptions I and II discussed above. By assuming that there are no collisions between walls and other robots and that there is no noise, we can completely ignore the infrared proximity sensors. As a consequence, the vector $\mathbf{I}(t)$ is determined only by ground and sound sensors. Ground sensors are completely characterised by the position of the robot in the environment. More precisely, the readings of the ground sensor depend on the y position and orientation θ of the robot over the grey gradient. It is therefore possible to determine a function that, given the robot position y , its orientation θ , and the

perceived sound s at time t , returns the vector of inputs $\mathbf{I}(t)$:

$$\mathbf{I}(t) = \mathcal{I}(y(t), \theta(t), s(t)) = \mathcal{I}(y, \theta, s)|_t. \quad (7)$$

Hereafter, we use the abbreviated notation $|_t$ to indicate that a certain variable or a set of variables is evaluated at time t .

The output vector $\mathbf{O}(t)$ is used to determine the speed of the robot treels and the status of the loudspeaker. Assumption III implies that the variation in position and orientation of the s -bot depends only on the speed of the two treels. It is therefore possible to determine a function that associates the output vector to the new position of the robot:

$$\langle y(t+1), \theta(t+1), S(t+1) \rangle = \mathcal{O}(\mathbf{O}(t)), \quad (8)$$

where $S(t)$ is the signal emitted at time t . The sound perception at time $t+1$ is determined by the emitted signal $S(t+1)$, together with the emitted signals of the other s -bots. In other words, s -bots are coupled by means of the global binary communication channel they are provided with, so that it is possible to determine the following coupling rule:

$$s(t) = \max_r S_r(t) \in \{0, 1\}, \quad (9)$$

which specifies that a binary signal is perceived if and only if at least one s -bot r is signalling. Note that the sound perception $s(t)$ is equal for all robots in the environment because communication is global and binary. When there is only one robot, the above coupling rule simplifies to $s(t) = S(t)$. Putting everything together, we obtain a discrete-time formulation of the behaviour of the individual robot:

$$\langle y, \theta, s \rangle|_{t+1} = \mathcal{O}(\mathcal{N}(\mathcal{I}(y, \theta, s)|_t, \mathbf{c})) = \mathcal{B}_{\mathbf{c}}(y, \theta, s)|_t. \quad (10)$$

presents various plots of the vector fields. The top-left 3D plot suggests how the state of an *s-bot* starting at any point in its phase space evolves over time. Together with the vector field, the continuous line indicates a closed orbit, due to the 2π -periodic boundary conditions of θ . This closed orbit appears to be a limit cycle attractor, as indicated by the convergence of all trajectories computed from starting positions extensively covering the phase space (data not shown). The existence of such a limit cycle attractor indicates that the individual behaviour produces a stable, periodic motion. The plots at the bottom of Figure 5 reveal the details of such periodic oscillations. An *s-bot* positioned at $\langle y, \theta, s \rangle \approx \langle 0, \frac{3}{2}\pi, 0 \rangle$ follows the directions indicated by the bottom-left vector field, which are parallel to the y axis: the robot moves on a straight line until $y \approx 0.75$. At this point, the vector field indicates that the trajectory jumps to plane $s = 1$, which corresponds to the *s-bot* signalling and therefore perceiving its own signal. The robot now follows the direction indicated by the bottom-right vector field, which corresponds to a clockwise rotation at constant y , followed by a circular trajectory. During this movement, the *s-bot* keeps on signalling until its trajectory jumps back to plane $s = 0$. Now, the robot moves straight again, crosses the x axis and performs an identical sequence of movements on the opposite side of the arena until it comes back to the initial position. It is worth noting the symmetry of the vector fields, which reflects the symmetry in the gradient painted on the arena floor.

Other important information can be extracted from the vector field: the signalling behaviour. For any state $\langle y, \theta, s \rangle$, the vector field indicates the variation of the perceived signal $\Delta s(y, \theta, s)$. By comparing the variation Δs for $s = 0$ and $s = 1$, it is possible to distinguish four different signalling behaviours:

1. no signal is emitted for all positions with $\Delta s(y, \theta, 0) = 0$ and $\Delta s(y, \theta, 1) = -1$.
2. a continuous signal is always emitted for all positions with $\Delta s(y, \theta, 0) = 1$ and $\Delta s(y, \theta, 1) = 0$. We refer to this behaviour as *environment-driven*

signalling, because it depends entirely on the position of the *s-bot* in the environment.

3. a continuous signal is emitted for all positions with $\Delta s(y, \theta, 0) = 0$ and $\Delta s(y, \theta, 1) = 0$, but only in response to a perceived signal. Otherwise, no signal is produced. We refer to this behaviour as *signal-driven* signalling.
4. an alternate signal is emitted for all positions with $\Delta s(y, \theta, 0) = 1$ and $\Delta s(y, \theta, 1) = -1$. In other words, signalling is driven by the *s-bot* position, but it is inhibited by the perception of a signal. As a consequence, the *s-bot* continuously switches the loudspeaker on and off. We refer to this behaviour as *alternate* signalling.

Figure 5 shows the signalling behaviour of c_8 in the top-right plot. Different signalling behaviours are indicated by filled circles of varying grey-levels. Notice that the limit cycle attractor traverses areas of the phase space characterised by varying signalling behaviour. A signal is produced when the *s-bot* enters the “environment-driven” area, and is stopped when the *s-bot* exits from the “signal-driven” area. Entering the signal-driven area having $s = 0$ does not lead to the emission of a signal, while entering with $s = 1$ maintains the previous signalling status.

In short, the behaviour of the *s-bot* is the result of the dynamics defined by two different vector fields, one characterised by no perceived signal ($s = 0$) and one characterised by a continuous signal ($s = 1$). In the latter case, the dynamics are characterised by a limit cycle attractor corresponding to the forced perception of a continuous signal (see the dotted line in the bottom-right plot of Figure 5). It is possible to notice how an *s-bot* moving along the normal limit cycle attractor approaches this “forced” attractor when $s = 1$. However, in approaching this attractor, the *s-bot* enters the “no signalling” area, and therefore it switches to movements dictated by the vector field for $s = 0$. Notice that in the latter case, an attractor does not exist within the range of possible values for y and θ . Nevertheless, the vector field is oriented such that the *s-bot*

encounters the signalling area again, and therefore switches back to movements towards the $s = 1$ attractor.

Controller c_{10} The individual behaviour produced by controller c_{10} —an example for the *reset mechanism* class—is presented in Figure 6. The 3D plot in the top-left of the figure shows the vector field and the limit cycle attractor that corresponds to the individual oscillatory behaviour. Looking at the vector field, it is possible to notice that an *s-bot* always emits a signal, which can be either alternate or environment-driven. As a consequence, the limit cycle attractor either lies in the $s = 1$ plane or it jumps back and forth between the two planes (i.e., the *s-bot* produces an alternate signalling pattern). The signalling behaviour for each position in the environment is better represented in the top-right plot, in which it is displayed together with a projection of the limit cycle attractor on the $y\theta$ plane. In this case, there exist only two areas with different signalling behaviour, which are anyway sufficient to support the synchronisation among *s-bots* as discussed in Section 4.3. To give an idea of the average direction of motion of the *s-bot* while perceiving an alternate signal, we plotted the average vectors obtained from the two vector fields given by $s = 1$ and $s = 0$ in the bottom-left part of Figure 6. On the bottom-right, instead, we show the vector field for $s = 1$. Looking at these plots, it is easy to describe the periodic oscillations of an *s-bot* as sequences of straight movements and anti-clockwise rotations. Also in this case, the behaviour of the *s-bot* is characterised by two vector fields corresponding to the perception of alternate and continuous signalling patterns. By forcing the perception of these patterns, it is possible to compute the attractors for alternate and continuous signalling. Alternate signalling leads to a limit cycle attractor, displayed as a dotted line in the bottom-left plot of Figure 6. Continuous signalling leads to two fixed point attractors displayed as crosses in the bottom-right plot of Figure 6. We can describe the *s-bot*'s oscillatory behaviour as the alternate convergence towards these attractors. By moving towards one attractor, the *s-bot* exits from the

corresponding signalling area and switches to movements towards the second attractor, which eventually lead the robot out of the second signalling area. This process generates a self-sustained oscillation. In particular, whenever an *s-bot* perceives a continuous signal, it moves towards a fixed point in the $y\theta$ space. The presence of such fixed points—which we refer to as the *reset configurations*—is a characterising feature of this class of behaviours and has a key role in the synchronisation mechanism, as we shall discuss in the following section.

4.3 Synchronisation Behaviour

In the previous section, we described the individual behaviour displayed by an *s-bot*. In particular, we focused on the stable oscillatory behaviour, ignoring the transitory phase that leads to the periodic motion. This transitory phase is relevant for the onset of synchronisation in groups of robots that influence each other until stable synchronous oscillations emerge. In this section, we exploit the dynamical system model introduced above to discuss the mechanisms that lead to synchrony. In particular, we use the formulation given in equation (11), which accounts for multiple robots coupled by a global binary communication channel. From equation (11), we observe that the *s-bots*' movements are governed solely by the individual behaviour \mathcal{B}_c , which was analysed in the previous section, and by the coupling rule (9), which states that a signal is perceived whenever any *s-bot* emits a signal. As a consequence, it is possible to describe the behaviour of synchronising *s-bots* by looking at how the individual movements change with respect to incoming signals. In the following, we give such a description for both controllers c_8 and c_{10} , for a system composed of 2 *s-bots*. The generalisation to R *s-bots* is the object of the scalability analysis (see Section 5).

Controller c_8 In order to describe the synchronisation for the *modulation mechanism* class of controllers, we analyse the transitory phase of two *s-bots* governed by c_8 while they entrain their oscillations. Figure 7 presents various

plots that represent different stages of the synchronisation. In the upper part, the position y for the two robots—as predicted by the mathematical model—is plotted with respect to time. It is possible to observe that after an initial transitory phase, the robots converge towards coordinated movements. In particular, the position y is “modulated” through communicative interactions: the robot that signals first influences the behaviour of the other robot, which anticipates the turnabout in response to the perceived signal. A better idea on how synchronisation is achieved is given by plotting the trajectories of the two robots over the vector fields for $s = 0$ and $s = 1$ (see the central and bottom plots of Figure 7). The two *s-bots* start in the points indicated by ‘O’, none of which is signalling. As a consequence, the *s-bots* follow the top-left vector field, until they reach the point indicated by an ‘A’. At this stage, one of the robots emits a signal (solid line), that triggers a behavioural change: the robots now follow the top-right vector field and both perform a clockwise turn. However, this rotation is not performed at the same speed: the robot at larger y (solid line) moves faster than the other (dotted line), as indicated by the size of the arrows of the vector field. The distance between the two robots is substantially reduced at this stage, which ends with the robots reaching the points indicated with ‘B’. In the interval from points ‘B’ to points ‘C’ no robot is signalling and no interaction is present. When the first robot enters the environment-driven signalling area (solid line), it again modifies the behaviour of the second robot (dotted line) by triggering an anticipated turnabout: the trajectories get closer to one another because of the difference in speed between a normal and an anticipated turnabout, eventually reaching point ‘D’ (see the bottom-left vector field), and with the same modulation mechanism the two trajectories nearly coincide by passing from points ‘E’ to ‘F’, as shown in the bottom-right vector field.

A formal analysis of the synchronisation behaviour can be performed exploiting the *Phase Response Curve* (PRC) associated with the individual oscillations. A PRC is obtained by delivering a precisely timed perturbation to an uncoupled oscillator, and measuring the effects on the oscillator period [52, 53]. More pre-

cisely, given the period T of the oscillator, a perturbation is applied at a phase $\phi = t_p/T$, where t_p is the beginning of the perturbation. Then, the perturbed period T_p is measured and a phase response $F(\phi) = (T - T_p)/T$ is calculated: a positive value represents a phase advance—i.e., the oscillator is pushed forward by the perturbation; conversely, a negative value represents a phase delay. In this work, we measure the free-running period T of the *s-bot*'s oscillatory behaviour as the time between two consecutive entries in the no-signalling area. We record the signalling pattern emitted during a single period, and we use such recording as perturbation, delayed by a phase ϕ as indicated in Figure 8 (at the top of the figure). Given that the perturbation lasts a full period T , we measure its effects asymptotically looking at the difference in phase between perturbed and free-running oscillations, once the perturbed oscillation has settled back on the limit cycle attractor [54]. The normalised PRC for controller c_8 is plotted in Figure 8 (at the bottom). On the one hand, for small $\phi < 0$ we observe a phase advance—i.e., whenever an *s-bot* perceives a signal shortly before signalling itself, the *s-bot* reacts by slightly anticipating the following signal emission. On the other hand, for $\phi > 0$ we observe a phase delay—i.e., whenever an *s-bot* perceives a signal after signalling itself, it reacts by delaying the following signal emission. In both cases, we observe a tendency to reduce the phase difference with the perceived signal by “modulating” the signal emission time. The onset of synchronisation is mainly the result of this modulation mechanism. Exact synchronisation is obtained due to the zero phase response observable for small positive ϕ , which allows a reciprocal fine-tuning of the phases between the coupled oscillators.

Controller c_{10} The *reset mechanism* class of controllers presents a slightly different synchronisation behaviour with respect to the modulation class. The main difference consists in the presence of fixed point attractors towards which the trajectories converge when an external signal is perceived. Figure 9 presents the synchronisation phase for two robots controlled by c_{10} . The upper plot shows

that the distances of the two robots initially approach each other, because of the fact that one of the robots keeps a constant position $y \approx 0.7$. After this initial phase, the robots start oscillating and rapidly achieve synchrony. The details of the reset mechanism can be assessed looking at the trajectories of the two robots plotted over the vector fields (see the central and bottom plots in Figure 9). In the initial phase, the robots move from points ‘O’ to points ‘A’. The robot with smaller y (dotted line) is placed in the environment-driven signalling area and it is therefore emitting a continuous tone (see the top-left vector field). The other robot (solid line) rapidly converges onto the fixed point attractor, i.e., the *reset configuration* indicated by a cross in the figure. It remains there as long as a continuous signal is perceived, i.e., until the second robot enters the alternate signalling area (see the points indicated by ‘A’). At this stage, the y position of the two robots is very close, but they present a large discrepancy in the orientation θ , which is reduced due to the different rotation speed between the robots. When the robots reach points ‘B’, they are very close to each other. At this point, continuous signalling starts again, and robots exchange roles: while approaching the points ‘C’, one robot (dotted line) moves towards the second fixed point attractor, slowing down in order to stop there, further reducing the distance from the other robot (solid line). When points ‘C’ are reached, alternate signalling starts again and the robots reduce the difference in their orientation due to slightly different rotation speed (see the bottom-left vector field). From points ‘D’ to ‘E’ the robots nearly converge to the same trajectory, eventually achieving synchronous movements at points ‘F’ and ‘G’.

Simplifying, we observed that synchronisation of the y position is achieved mainly through convergence toward the *reset configuration*, which is maintained as long as a continuous signal is perceived. Refinements still take place during the oscillatory motion due to a reduced velocity of the robots while approaching or leaving the *reset configuration*. This is also evident looking at the PRC corresponding to this behaviour (see Figure 10), which was computed in the

same way as for controller c_8 , taking as reference point for the identification of the free-running period the entry in the alternate signalling area. In this case, for both $\phi > 0$ and $\phi < 0$ a phase delay is observed, which is a result of the slow convergence of the *s-bot* toward the fixed point attractor described above. This behaviour can be interpreted as the *s-bot* consistently trying to delay its signal emission. Synchrony arises from the mutual interactions of two coupled oscillators that always try to signal the latest. In fact, as observed above, the *s-bots* switch role at each iteration until synchronisation is achieved. It is interesting to notice that a similar mechanism is considered at the basis of chorusing in many animal species, where synchrony results from the attempt of each individual to anticipate the signal of its neighbours [30].

5 Scalability Analysis

So far, we have given a detailed description of the synchronisation mechanism employed by the two classes of evolved controllers, i.e., the *modulation mechanism* and the *reset mechanism*. We have described the onset of synchronisation giving examples with two *s-bots* only, but no prove is given that the described mechanisms scale to a larger number of robots. In this section, we provide a scalability analysis that aims at testing all evolved behaviours with a varying number of robots. We first test the evolved behaviours in the original simulation environment, which includes all the features that were neglected in the dynamical system model (e.g., physical interactions and environmental noise, see Section 5.1). Then, the scalability of the synchronisation mechanism *per se* is presented in Section 5.2, and predictions from the mathematical model are discussed in Section 5.3.

5.1 Scalability of the Evolved Behaviours

In order to establish to what extent the evolved behaviours function with increasing group size, we performed a series of tests evaluating the performance

of the evolved behaviour with groups of 3, 6, 12, 24, 48 and 96 *s-bots*. Tests are performed with the same simulation environment used for the evolutionary optimisation, and the performance is measured according to the description given in Section 2.3, without any modification. In order to ensure a fair comparison in different trials, we decided to keep a constant, uniform density of robots in the arena. In fact, in a crowded situation, the ability to synchronise would be disrupted by frequent physical interferences—i.e., evolved collision avoidance behaviour—among robots and between robots and walls. By ensuring a constant initial density we limit the negative effects of overcrowding and are able to compare the performance of the system with varying group size. In order to keep a constant robot density equal to the one used in the evolutionary experiments, that is, 0.25 robots per square meter, we lengthened the arena in the x direction. So, for instance, a group size of 96 robots corresponds to an arena with a length of 192 m . Despite the increased arena length, we keep the same communication protocol, i.e., communication continues to be binary and global, with all robots affecting each other.⁵ This choice allows us to evaluate the scalability of an evolved behaviour without modifying the features of the communication channel. In this way, we hope to understand under which conditions behaviours evolved with three robots scale to large groups.

Using the above setup, we evaluated all best evolved controllers 100 times for each group size. The obtained results are presented in Figure 11. It is possible to notice that most of the best evolved controllers have a good performance for groups composed of 6 *s-bots*. Performance degrades for larger group sizes and only a few controllers produce scalable behaviours up to groups formed by 96 *s-bots*. The main problem that reduces the scalability of the evolved controllers is given by the higher probability per time step of physical interactions among robots: in fact, the larger the group size, the more probable the encounters

⁵With respect to real world scenarios, this choice may seem unrealistic. However, other communication modalities than sound signalling may be employed to provide physical robots with a global, binary communication protocol. For instance, *s-bots* are provided with a WiFi interface, which could be employed to send and receive messages, independent of the distance among the robots.

among robots per time step, despite the constant initial density we introduced. This is confirmed by the higher number of collisions detected with larger groups. Recall that a null performance is assigned to each trial that terminates because *s-bots* did not manage to avoid each other (see Section 2.3). For some controllers, such as c_3 and c_4 , the number of trials terminated due to collisions increases with the group size, as is evident in Figure 11. Nevertheless, whenever *s-bots* successfully avoid collisions, their dodging movements provoke a temporary de-synchronisation of at least two robots, which have to re-gain synchronisation. Considering this, the reduction in performance observed for large groups can be explained by the following reasons:

- the higher the group size, the longer the transitory phase that leads to synchronous oscillations. During the transitory phase, physical interactions among *s-bots* are more probable because the robots have not yet settled into stable oscillations;
- the computation of the synchronisation component F_S conservatively chooses the minimum cross correlation among all *s-bot* pairs (see equation (5)). As a consequence, even a few physical interactions may lead to a strong decrease in performance;
- global and binary communication implies that the whole group is influenced by the attempt of a few robots to re-gain synchronisation. In other words, the movements of all *s-bots* may be influenced by the attempt of a single *s-bot* to synchronise.

All the above reasons contribute to reduce the scalability of the evolved controllers. In addition, another interference among robots limits the scalability. This is a communicative interference that does not allow the robots to interact through sound signals, and therefore to synchronise. We discuss this problem in the following section, which is dedicated to the scalability of the evolved synchronisation mechanisms.

5.2 Scalability of the Synchronisation Mechanism

In this section, we analyse the evolved controllers in order to uncover the effect of an increasing group size over the synchronisation mechanisms described in Section 4. To do so, we modify the simulation environment in order to neglect the physical interactions among robots and between robots and walls, whose influence has been discussed in the previous section. Noise is simulated as described in Section 2. Using the same experimental setup described above, we perform an analysis for different group sizes focusing on the synchronisation aspect only. The obtained results are plotted in Figure 12. Differently from what was observed above, in this case many controllers present very good scalability, with only a slight decrease in performance due to the longer time required by larger groups to exactly synchronise (see controllers c_2 , c_8 , c_{10} , c_{12} , c_{14} , c_{18} and c_{19}). Controllers c_5 and c_6 present good scalability, but are characterised by a potential loss in performance due to slow convergence to synchrony of large groups. This result confirms the analysis given in the previous section about the negative impact of physical interferences among *s-bots*. In fact, removing the necessity to avoid collisions leads to scalable behaviours.

Nevertheless, many other controllers present poor scalability properties. In these cases, the performance presents a high variance up to a certain group size. Then, the performance stabilises at a low, constant value, independent from the initial conditions and the number of robots used. This value, which is characteristic of each non-scaling controller, represents the performance of an *incoherent* behaviour of the robots. In other words, for every initial condition we tested, all robots converged to a stable behaviour without being capable of synchronising with any other robot. By observing the actual behaviour produced by these controllers, we realised that the incoherent state is caused by a communicative interference problem: the signals emitted by different *s-bots* overlap in time and are perceived as a *fixed signalling pattern*, either continuous or alternate (recall that the sound signals are global and that they are perceived in a binary way, preventing an *s-bot* from recognising different signal sources).

If the perceived signal does not vary in time, it does not bring enough information to be exploited for synchronisation. This problem is the result of the fact that we used a “global” communication form in which the signal emitted by an *s-bot* is perceived by any other *s-bot* anywhere in the arena. Moreover, from the perception point of view, there is no difference between a single *s-bot* and a thousand signalling at the same time. The lack of locality and of additivity is the main cause of failure for the scalability of the evolved synchronisation mechanisms. However, as we have seen, this problem affects only some of the analysed controllers. For the remaining controllers, the evolved communication strategies present a very good scalability that is only weakly influenced by the group size. A discussion about the causal relationship between the individual behaviour and scalability is given in the following section.

5.3 Predictions of the Mathematical Model

Given the individual behaviour, is it possible to predict whether the synchronisation mechanism is scalable? What is the minimum group size that presents the interference problem? In this section, we try to answer these questions exploiting the mathematical model introduced in Section 4.1.

We start from the observation that, if a synchronisation mechanism does not scale with the group size R , there should exist an *incoherent attractor* in which the system converges, alternative to the synchronous one. In other words, the dynamical system (11) undergoes a bifurcation with varying parameter R , so that two attractors are observable: the synchronous and the incoherent one. In order to predict from the individual behaviour whether such a bifurcation exists or not, it is necessary to understand the conditions for the existence of an incoherent attractor.

In the previous section, we empirically observed that, whenever an evolved synchronisation mechanism does not scale, robots perceive a fixed signalling pattern, either continuous or alternate. In such a situation, the *s-bots* do not receive information about the position and orientation of other robots: each *s-bot*

broadcasts such information, which however gets lost due to the communicative interference we discussed above. It is easy to prove that, given a fixed signalling pattern $s(t+1) = f(s(t))$, no synchronisation is possible. If an *s-bot*, r , perceives a fixed signalling pattern, its behaviour is not influenced by other *s-bots*, and can be predicted as follows:

$$\begin{cases} \langle y_r, \theta_r, S_r \rangle|_{t+1} = \mathcal{B}_c(y_r, \theta_r, s)|_t \\ s|_{t+1} = f(s)|_t \end{cases} . \quad (12)$$

It is therefore possible to study the above behaviour, and analyse possible attractors—be they fixed points or limit cycles. If no attractor exists, *s-bots* movements diverge and no synchronisation can be observed. If the attractors exist and are fixed points, robots do not move in a synchronous manner because they do not move at all. If the attractor is a limit cycle, the position on the cycle only depends on the initial position of the robot, so that *s-bots* starting from different initial positions will not synchronise.

In summary, if a fixed signalling pattern is perceived, *s-bots* cannot achieve synchronisation. In other words, a fixed signalling pattern is a sufficient condition for the existence of an incoherent attractor. As a consequence, artificial evolution shaped the individual behaviours in order to avoid fixed signalling patterns, and each evolved controller produces at least two different signalling behaviours. However, the communicative interference highlighted before can lead to the perception of a fixed signalling pattern which is self-sustained by the group, even if each individual *s-bot* varies its own signalling behaviour. By analysing the individual behaviour in equation (12) for the signalling patterns produced by the same evolved controller,⁶ it is possible to define the conditions for the existence of communicative interference, which corresponds to the existence of an incoherent attractor. In particular, we claim that if equa-

⁶For a given controller, it is sufficient to consider the signalling pattern that subsumes all other possible patterns. For instance, in c_{10} , the continuous signalling pattern $f(s) = 1$ subsumes the alternate signalling $f(s) = 1 - s$, so that the communicative interference can be created only by the former.

tion (12) presents attractors that are contained within the region of phase space in which $S_r|_{t+1} = f(s)|_t$ —which we refer to as the *non-interaction area*—then for a sufficiently large R the group can produce a fixed signalling pattern. As a consequence, an incoherent attractor exists and the evolved synchronisation mechanism is not scalable.

To prove the above claim, suppose that an *s-bot* perceives a fixed signalling pattern $f(s)$, which belongs to its repertoire of signalling behaviours. Its movements are therefore described by equation (12). By hypothesis, the attractors of this system are contained at least partially within the *non-interaction area*, so that:

$$\exists t : S_r|_{t+1} = \mathcal{B}_c^S(y_r, \theta_r, f(s))|_t = f(s)|_t, \quad (13)$$

where \mathcal{B}_c^S indicates the component of \mathcal{B}_c related to the production of the signal S . In other words, *s-bot* r produces the same signalling pattern $f(s)$ while traversing the *non-interaction area*. For a sufficiently large R , the self-production of a fixed signalling pattern becomes possible:

$$\forall t \exists r : S_r|_{t+1} = \mathcal{B}_c^S(y_r, \theta_r, f(s))|_t = f(s)|_t, \quad (14)$$

that is, at every instant t there is an *s-bot* r that contributes in producing the signalling pattern.

To summarise, the analysis of the attractors of the individual behaviour under the forced perception of a fixed signalling pattern reveals whether such signalling patterns can be self-sustained by a sufficiently large group. If this is the case, an incoherent attractor exists for the system in equation (11) and the evolved behaviour does not scale. Otherwise, scalability is possible. The controllers analysed in Section 4.2 produce both a scalable synchronisation mechanism, as observed in Figure 12. In fact, both present attractors for a fixed signalling pattern that lie outside of the corresponding *non-interaction area*. Controller c_8 is characterised by a limit cycle attractor shown as a dotted line in the bottom-right vector field of Figure 5, which is completely contained in

the no-signalling area. Controller c_{10} presents two fixed point attractors, shown as crosses in the bottom-right vector field of Figure 6, which lie outside of the environment-driven signalling area.

To further prove our claims, we analysed the individual behaviour of the best performing controller that does not present scalability, namely c_{13} . The evolved behaviour can be analysed with the 3D vector field of Figure 13 that shows the individual behaviour under normal conditions. The right vector field in Figure 13 corresponds to the behaviour of the *s-bot* in presence of fixed continuous signalling $f(s) = 1$. It is possible to notice that the limit cycle attractor for this condition traverses the environment-driven signalling area. As a consequence, with a sufficiently large number of *s-bots* the evolved synchronisation mechanism does not scale, as can be seen in Figure 12. The bifurcation of the corresponding dynamical system with the number of robots R is well represented in Figure 14. Here, we measured the average standard deviation of the absolute value $|y|$ in 100 trials for varying group size. The points in Figure 14 represent the average value, and the black line represents the least square error fitting function. It is possible to notice that for small R , only the synchronous attractor exists, and the robots always achieve a zero standard deviation in the $|y|$ position. For larger group sizes, the incoherent attractor appears, and for $R > 40$ the system always converges to the incoherent attractor. We can describe this situation from a dynamical system perspective. With increasing group size, the system undergoes a bifurcation, so that, on the one hand, the larger the group size R , the larger the basin of the incoherent attractor, which becomes more and more probable. On the other hand, the synchronous attractor does not disappear: if robots start synchronised, they will keep synchrony for ever. However, the basin of the synchronous attractor shrinks with increasing R , making it less probable to observe the onset of synchronisation for large groups.

A further prediction from the mathematical model consists in the minimum group size, R_m , for which the incoherent attractor exists. This group size depends on the time each robot spends in the *non-interaction area* while moving

over the limit cycle for the corresponding fixed signalling pattern. In fact, in order to satisfy condition (14), it is necessary that while a robot moves within the *non-interaction area*, another robot prepares to enter the *non-interaction area*. In other words, R_m robots should be evenly spaced over the limit cycle so that, when one *s-bot* exits the *non-interaction area*, another one enters it, therefore sustaining the production of the fixed signalling pattern. As a consequence, the minimum group size R_m is given by:

$$R_m = \left\lceil \frac{T}{T_n} \right\rceil, \quad (15)$$

where T is the period of a single oscillation, and T_n is the fraction of this period spent within the *non-interaction area* while moving on the limit cycle. For controller c_{13} , we experimentally obtained $R_m = 6$, which can be considered a theoretical lower bound for the minimum group size. We actually observed the existence of the incoherent attractor for a minimum group size of 9, as shown in Figure 14.

A final remark about scalability concerns noisy conditions: in the presence of environmental noise, the trajectories of the robots given by equation (12) may oscillate around the attractors. In such conditions, the noisy trajectories may fall into the *non-interaction area* even if the attractors do not lie within it, but are sufficiently close. With a certain probability—depending on the group size, the noise level and the vicinity of the attractors to the *non-interaction area*—the system may converge towards an incoherent attractor caused by environmental noise.

6 Tests with Physical Robots

So far, we have shown how artificial evolution can synthesise efficient and scalable synchronisation mechanisms which are based on minimal communication strategies. In this section, we test the evolved behaviour with physical robots. Such tests should be considered a *proof-of-concept* about the usage of evolution-

ary techniques for the synthesis of swarm robotics controllers. Our main interest is therefore testing to what extent the behaviours evolved in simulations transfer to physical robots, and identifying the possible causes of failure.

For tests with physical robots, we chose controller c_8 , as it presents both a high performance and good scalability properties. The neural network controller is used on the physical *s-bots* exactly in the same way as in the simulations. The sensor readings are taken every 100 *ms*, are scaled to the range [0,1] and finally fed to the neural network. The outputs of the network are used to control the wheels and the loudspeaker. Due to limited room, we reduced the x size of the experimental arena to 3 *m*. Within the arena, we defined the initial positions of the robots by tracing a grid of 11×7 evenly spaced points, 25 *cm* distant from each other. We initialised the robots by randomly choosing, without replacement, the position within the grid and by choosing the orientation randomly from 8 possible angles. In order to compute the performance of the evolved solutions, we could not use the fitness function defined in Section 2.3, because technical reasons prevented the use of an overhead camera to obtain a reliable measure of the *s-bots*' position within the arena, necessary to compute both F_M and F_S . Instead, we focused on synchronisation only and we used the perception of the ground sensors to estimate the position y over the gradient. Using this estimate, we computed the average cross-correlation among the time series recorded on the *s-bots* for each trial.

Using the above setup, we performed 20 trials with 2 *s-bots*, and 20 trials with 3 *s-bots*.⁷ In order to make a fair comparison between the performance of the physical robots and the one obtained in simulation, we performed an equal number of trials in simulation, starting from identical initial positions in an identical arena, and we computed the same performance measure that was used for the tests with physical robots. The obtained results are plotted in Figure 15. Tests with 2 robots present a fairly good performance in some trials, with *s-bots* displaying the ability to achieve synchronised movements, to avoid

⁷Videos of the experiments with physical robots can be found at <http://lara1.istc.cnr.it/esm/trianni-nolfi-ieee09>.

collisions and to recover synchrony. Other trials present lower performance: in these cases, *s-bots* display a tendency to synchronise, which is however disrupted by collision avoidance manoeuvres and by noise. Concerning tests with 3 *s-bots*, a generalised reduction in performance with respect to simulation was observed. Notice however that the simulated experiments also present a lower performance if compared with tests performed with 2 robots. In fact, in reality as well as in simulation, collision avoidance is more frequent due to the reduced size of the experimental arena, and therefore synchrony is often disrupted by dodging manoeuvres. Additionally, we observed that in reality it takes longer than in simulation to manoeuvre out of the collision condition. Finally, real-world experiments have been conducted under severe conditions regarding ground sensors' noise. In fact, the trees of the *s-bots* continuously produce small bumps that result in very noisy readings. We tested the effects of increased noise in the simulations, and we found behavioural patterns similar to the real-world scenario (data not shown). We consider this as an indication that, besides the increased difficulty in manoeuvring out of a collision condition, the low performance observed in reality should be ascribed to increased noise in the ground sensors.

7 Conclusion

Much like natural evolution produced swarms of fireflies able to self-organise to achieve coherent group behaviour, artificial evolution can synthesise self-organising swarms of robots that accomplish complex tasks. In this respect, swarm intelligence can benefit from the study and analysis of natural as well as artificial systems: in both cases, a deep understanding of the dynamics that govern the individual behaviour and the social interactions can underpin novel developments in the engineering of swarm intelligent systems. In this paper, we have presented an artificial evolutionary process designed to shape the behaviour of a robot system to display self-organised synchronisation. We have also shown

how the dynamical system analysis can explain the evolved mechanisms and predict the behaviour of the robot system for varying group size. We believe that this analysis can bring useful insights on how to build—through automatic techniques or hand-design—swarm robotics systems that are capable of self-organised synchronisation and that scale to a large number of robots. In fact, we have given a clear description of the building blocks necessary to produce synchronised behaviours, and most importantly we have decoded the individual behaviour to find the conditions that allow the system as a whole to synchronise, independent of the group size.

Concerning the generality of the proposed approach, it is important to notice that the dynamical system analysis performed in this work depends neither on the type of controller nor on the methodology used to design it. In fact, the mathematical model introduced in Section 4.1 treats the controller as a black box, while the design methodology is not even considered. The only knowledge about the controller we exploited refers to the absence of internal states. Otherwise, we simply identified the state variables of the system and made all the simplifications necessary to perform the numerical analysis. Therefore, similar dynamical models could be developed for different experimental settings and different swarm robotics contexts, for simulated or physical robots, in order to uncover the mechanisms that lead to certain group behaviour, and possibly to predict emergent features of the system, much like we did with the scalability analysis presented in Section 5.3.

The synthesis of collective, coordinated behaviours in physical robots is not a trivial enterprise. In fact, due to the indirect relationship between simple local rules and the system’s global properties, the definition of the individual behaviours is particularly challenging [55]. The results presented in this paper support the use of evolutionary robotics techniques for the development of self-organising behaviours. The evolved behaviours feature high efficiency and scalability properties, and have been positively tested with physical robots. To obtain these results, we adopted a minimal approach that does not postulate

the need of internal dynamics for the robots to be able to synchronise. Instead, we stress the importance of the dynamical coupling between robots and environment. Robots can be described as *embodied oscillators*, their behaviour being characterised by a period and a phase. In this perspective, the movements of an *s-bot* correspond to advancements of the oscillation phase. Robots can modulate their oscillations simply by moving in the environment and by modifying their dynamical relationship with it. Such modulations are brought forth in response to the perceived communication signals, which also depend on the dynamical relationship between the *s-bot* and the environment. In this way, simple and reactive behavioural and communication strategies are sufficient to implement effective synchronisation mechanisms. We have also analysed the scalability of the evolved controllers, showing that synchronisation can be obtained in large groups, even though large groups were never tested during the evolutionary optimisation (controllers were evolved always using three *s-bots*). We observed that physical or communicative interferences may prevent the system from synchronising. This is a consequence of the global and binary communication channel we used, which results in an excessive influence of the signal emitted by any single individual on the dynamics of the whole group. Better results may be achieved exploiting a *local* and *additive* communication, which ensures that only the signals emitted by neighbours would be taken into account and that the number of contemporary signalling robots is relevant. We will account for this possibility in future work. Another interesting issue to study in future developments is to provide evolving robots with the possibility to discriminate between self-produced and external signals. Such an ability may be based either on an additional sensory channel that detects only the self-produced signal, or on the internal dynamics of the neural controller, which could evolve some neural mechanism similar to the corollary discharge observed in some insects [56].

By looking at the evolved behaviours, we recognised two different strategies. The analysis of their dynamics indicates that the two strategies are based on the same general mechanism: the tendency of the robots to move towards the

attractor that corresponds to a certain signalling pattern—continuous or alternate. The difference between the two strategies can be recognised in the type of attractors: the one based on the so-called *modulation mechanism* features a limit cycle attractor, while the one based on the *reset mechanism* is characterised by fixed point attractors. Similar mechanisms are also observed in biological oscillators. For instance, different species of fireflies present different synchronisation mechanisms, based on delayed or advanced phase responses [4,5,18]. Moreover, our results seem to be in accordance with the precedence effect explanation of chorusing, for which synchronous signalling evolved as a result of the attempt of each individual to anticipate the signal of its neighbours [30]. In future work, we plan to extract further results from the analytical model, in order to uncover the details about the evolved synchronisation mechanisms and further study our artificial system in comparison with biological examples.

In conclusion, we believe that studies about synchronisation such as the one presented in this paper, notwithstanding the explicitly simplified experimental setup, can have a strong impact on future studies in swarm robotics. In our work, robots oscillate by exploiting environmental cues, and synchronise on the basis of communication only. We can imagine a swarm robotics system in which each individual robot behaves according to the environmental contingencies it experiences, and in parallel synchronises with other robots creating assemblies of coherent activity that lead to the achievement of a collective goal. We used the term *assembly*, commonly found in cognitive neurosciences [6,9], not accidentally: much like neurons in the brain synchronise to bring forth complex cognitive functions, robots that synchronise are dynamically coupled and can form groups performing coordinated, cooperative activities. The synchronised robots collectively identify themselves as a cooperative unit and potentially differentiate from other units—either behaviourally or physically through a self-assembling process [51]. Such differentiation is brought forth on the basis of the environmental contingencies experienced by the robots, and on the basis of communicative interactions. In this way, we can imagine that, in a swarm

robotics system, allocation of roles and tasks arises as a result of the dynamical coupling among robots and between robots and the environment.

Acknowledgement

This work was supported by the Swarmanoid project, funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission under grant IST-022888. The information provided is the sole responsibility of the authors and does not reflect the European Commission's opinion. The European Commission is not responsible for any use that might be made of data appearing in this publication. The authors thank Paola Pellegrini and Mauro Birattari for providing the tools necessary to perform the statistical analysis. Thanks to Guido De Croon, Shervin Nouyan and Federico Vicentini for the useful discussions about the dynamical system analysis. The authors also wish to thank the anonymous reviewers for their insightful comments.

References

- [1] S. H. Strogatz, *Sync: The Emerging Science of Spontaneous Order*. Hyperion Press, New York, NY, 2003.
- [2] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge University Press, Cambridge, UK, 2001.
- [3] M. Bennett, M. F. Schatz, H. Rockwood, and K. Wiesenfeld, "Huygens's clocks," *Proceedings of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, vol. 458, no. 2019, pp. 563–579, 2002.
- [4] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.

- [5] S. H. Strogatz and I. Stewart, “Coupled oscillators and biological synchronization,” *Scientific American*, vol. 269, no. 6, pp. 102–109, 1993.
- [6] P. Fries, “A mechanism for cognitive dynamics: neuronal communication through neuronal coherence,” *Trends in Cognitive Sciences*, vol. 9, no. 10, pp. 474–480, 2005.
- [7] C. von der Malsburg, “The what and why of binding: The modeler’s perspective,” *Neuron*, vol. 24, no. 1, pp. 95–104, 1999.
- [8] W. Singer, “Neuronal synchrony: A versatile code for the definition of relations?” *Neuron*, vol. 24, no. 1, pp. 49–65, 1999.
- [9] A. K. Engel, P. Fries, and W. Singer, “Dynamic predictions: Oscillations and synchrony in top-down processing,” *Nature Reviews Neuroscience*, vol. 2, pp. 704–716, 2001.
- [10] T. J. Sejnowski and O. Paulsen, “Network oscillations: Emerging computational principles,” *The Journal of Neuroscience*, vol. 26, no. 6, pp. 1673–1676, 2006.
- [11] O. Jensen, J. Kaiser, and J.-P. Lachaux, “Human gamma-frequency oscillations associated with attention and memory,” *Trends in Neurosciences*, vol. 30, no. 7, 2007.
- [12] M. Bartos, I. Vida, and P. Jonas, “Synaptic mechanisms of synchronized gamma oscillations in inhibitory interneuron networks,” *Nature Reviews Neuroscience*, vol. 8, pp. 45–56, 2007.
- [13] S. Camazine, J.-L. Deneubourg, N. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*. Princeton University Press, Princeton, NJ, 2001.
- [14] M. J. Ryan, M. D. Tuttle, and L. K. Taft, “The costs and benefits of frog chorusing behavior,” *Behavioral Ecology and Sociobiology*, vol. 8, no. 4, pp. 273–278, 1981.

- [15] M. D. Greenfield and K. C. Shaw, “Adaptive significance of chorusing with special reference to the orthoptera,” in *Orthopteran Mating Systems: Sexual Competition in a Diverse Group of Insects*, D. T. Gwynne and G. K. Morris, Eds. Westview Press, Boulder, Colorado, 1983, pp. 1–27.
- [16] J. S. Kotiaho, R. V. Alatalo, J. Mappes, and S. Parri, “Adaptive significance of synchronous chorusing in an acoustically signalling wolf spider,” *Proceedings of the Royal Society of London, Series B: Biological Sciences*, vol. 271, pp. 1847–1850, 2004.
- [17] B. Merker, “Synchronous chorusing and human origins,” in *The Origins of Music*, N. L. Wallin, B. Merker, and S. Brown, Eds. The MIT Press, Cambridge, MA, 1999, pp. 315–328.
- [18] J. Buck, “Synchronous rhythmic flashing of fireflies. II,” *The Quarterly Review of Biology*, vol. 63, no. 3, pp. 256–289, 1988.
- [19] P. Backwell, M. Jennions, N. Passmore, and J. Christy, “Synchronized courtship in fiddler crabs,” *Nature*, vol. 391, pp. 31–32, 1998.
- [20] J. R. Huguenard and D. A. McCormick, “Thalamic synchrony and dynamic regulation of global forebrain oscillations,” *Trends in Neurosciences*, vol. 30, no. 7, pp. 350–356, 2007.
- [21] C. Hammond, H. Bergman, and P. Brown, “Pathological synchronization in parkinson’s disease: networks, models and treatments,” *Trends in Neurosciences*, vol. 30, no. 7, pp. 357–364, 2007.
- [22] K. M. Spencer, P. G. Nestor, M. A. Niznikiewicz, D. F. Salisbury, M. E. Shenton, and R. W. McCarley, “Abnormal neural synchrony in schizophrenia,” *The Journal of Neuroscience*, vol. 23, no. 19, pp. 7407–7411, 2003.
- [23] S. H. Strogatz, D. M. Abrams, A. McRobie, B. Eckhardt, and E. Ott, “Crowd synchrony on the millennium bridge,” *Nature*, vol. 438, pp. 43–44, 2005.

- [24] A. Pikovsky, M. Rosenblum, and J. Kurths, “Phase synchronization in regular and chaotic systems,” *International Journal of Bifurcations and Chaos*, vol. 10, no. 10, pp. 2291–2305, 2000.
- [25] M. Rosenblum, A. Pikovsky, J. Kurths, C. Schaefer, and P. Tass, “Phase synchronization: from theory to data analysis,” in *Neuro-informatics and Neural Modelling*, ser. Handbook of Biological Physics, F. Moss and S. Gielen, Eds. Elsevier, Amsterdam, The Netherlands, 2001, vol. 4, ch. 9, pp. 279–321.
- [26] B. Boashash, “Estimating and interpreting the instantaneous frequency of a signal. I. Fundamentals,” *Proceedings of the IEEE*, vol. 80, no. 4, pp. 520–538, 1992.
- [27] —, “Estimating and interpreting the instantaneous frequency of a signal. II. Algorithms and applications,” *Proceedings of the IEEE*, vol. 80, no. 4, pp. 540–568, 1992.
- [28] Y. Kuramoto, “Phase dynamics of weakly unstable periodic structures,” *Progress of Theoretical Physics*, vol. 71, no. 6, pp. 1182–1196, 1984.
- [29] S. H. Strogatz, “From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators,” *Physica D: Nonlinear Phenomena*, vol. 143, no. 14, pp. 1–20, 2000.
- [30] M. D. Greenfield, M. K. Tourtellot, and W. A. Snedden, “Precedence effects and the evolution of chorusing,” *Proceedings of the Royal Society of London, Series B: Biological Sciences*, vol. 264, no. 1386, pp. 1355–1361, 1997.
- [31] H. Nijmeijer and A. Rodriguez-Abrego, *Synchronization of mechanical systems*. World Scientific Publishing Co. Pte. Ltd., 2003.
- [32] M. Mazzapoda and S. Nolfi, “Synchronization and gait adaptation in evolving hexapod robots,” in *From animals to animats 9: Proceedings of the Ninth International Conference on Simulation of Adaptive Behaviour*, ser.

- LNAI, S. N. et al., Ed., vol. 4095. Springer Verlag, Berlin, Germany, 2006, pp. 113–125.
- [33] M. Dorigo and E. Şahin, “Swarm robotics — special issue editorial,” *Autonomous Robots*, vol. 17, no. 2–3, pp. 111–113, 2004.
- [34] O. Holland and C. Melhuish, “An interactive method for controlling group size in multiple mobile robot systems,” in *Proceedings of the 8th International Conference on Advanced Robotics (ICAR '97)*. IEEE Press, Piscataway, NJ, 1997, pp. 201–206.
- [35] C. Melhuish, O. Holland, and S. Hoddell, “Convoying: using chorusing to form travelling groups of minimal agents,” *Robotics and Autonomous Systems*, vol. 28, pp. 207–216, 1999.
- [36] S. Wischmann, M. Huelse, J. F. Knabe, and F. Pasemann, “Synchronization of internal neural rhythms in multi-robotic systems,” *Adaptive Behavior*, vol. 14, no. 2, pp. 117–127, 2006.
- [37] M. Hartbauer and H. Römer, “A novel distributed swarm control strategy based on coupled signal oscillators,” *Bioinspiration and Biomimetics*, vol. 2, pp. 42–56, 2007.
- [38] S. Wischmann and F. Pasemann, “The emergence of communication by evolving dynamical systems,” in *From animals to animats 9: Proceedings of the Ninth International Conference on Simulation of Adaptive Behaviour*, ser. LNAI, S. N. et al., Ed., vol. 4095. Springer Verlag, Berlin, Germany, 2006, pp. 777–788.
- [39] R. Pfeifer and C. Scheier, “Sensory-motor coordination: The metaphor and beyond,” *Robotics and Autonomous Systems*, vol. 20, no. 2–4, pp. 157–178, 1997.

- [40] S. Nolfi and D. Floreano, *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press/Bradford Books, Cambridge, MA, 2000.
- [41] I. Harvey, E. A. Di Paolo, R. Wood, M. Quinn, and E. Tuci, “Evolutionary robotics: A new scientific tool for studying cognition,” *Artificial Life*, vol. 11, no. 1–2, pp. 79–98, 2005.
- [42] S. H. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Westview Press, Cambridge, MA, 1994.
- [43] R. D. Beer, “A dynamical systems perspective on agent-environment interaction,” *Artificial Intelligence*, vol. 72, pp. 173–215, 1995.
- [44] —, “Dynamical approaches to cognitive science,” *Trends in Cognitive Sciences*, vol. 4, no. 3, pp. 91–99, 2000.
- [45] G. Schöner, “Dynamical systems approaches to neural systems and behavior,” in *International Encyclopedia of the Social and Behavioral Sciences*, N. J. Smelser and P. B. Baltes, Eds. Pergamon Press, Oxford, UK, 2002, pp. 10571–10575.
- [46] E. Thelen, G. Schöner, C. Scheier, and L. B. Smith, “The dynamics of embodiment: A field theory of infant perseverative reaching,” *Behavioral and Brain Sciences*, vol. 24, no. 1, pp. 1–34, 2001.
- [47] P. Fitzpatrick, R. C. Schmidt, and C. Carello, “Dynamical patterns in clapping behavior,” *Journal of Experimental Psychology: Human Perception and Performance*, vol. 22, no. 3, pp. 707–724, 1996.
- [48] G. Schöner, “Timing, clocks, and dynamical systems,” *Brain and Cognition*, vol. 48, no. 1, pp. 31–51, 2002.
- [49] M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M.

- Gambardella, “Evolving self-organizing behaviors for a *swarm-bot*,” *Autonomous Robots*, vol. 17, no. 2–3, pp. 223–245, 2004.
- [50] F. Mondada, G. C. Pettinaro, A. Guignard, I. V. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, “SWARM-BOT: A new distributed robotic concept,” *Autonomous Robots*, vol. 17, no. 2–3, pp. 193–221, 2004.
- [51] E. Tuci, R. Groß, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo, “Cooperation through self-assembling in multi-robot systems,” *ACM Transactions on Autonomous and Adaptive Systems*, vol. 1, no. 2, pp. 115–150, 2006.
- [52] C. C. Canavier, R. J. Butera, R. O. Dror, D. A. Baxter, J. W. Clark, and J. H. Byrne, “Phase response characteristics of model neurons determine which patterns are expressed in a ring circuit model of gait generation,” *Biological Cybernetics*, vol. 77, pp. 367–380, 1997.
- [53] L. Glass and M. Mackey, *From Clocks to Chaos: the Rhythms of Life*. Princeton University Press, Princeton, NJ, 1988.
- [54] E. Brown, J. Moehlis, and P. Holmes, “On the phase reduction and response dynamics of neural oscillator populations,” *Neural Computation*, vol. 16, pp. 673–715, 2004.
- [55] V. Trianni, S. Nolfi, and M. Dorigo, “Evolution, self-organisation and swarm robotics,” in *Swarm Intelligence. Introduction and Applications*, ser. Natural Computing Series, C. Blum and D. Merkle, Eds. Springer Verlag, Berlin, Germany, 2008.
- [56] J. Poulet and B. Hedwig, “The cellular basis of a corollary discharge,” *Science*, vol. 311, no. 5760, pp. 518–522, 2006.

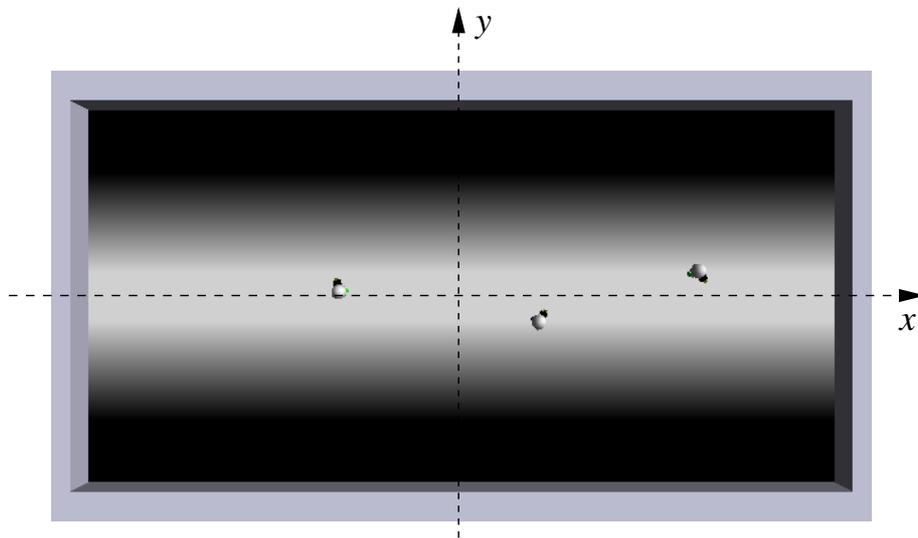


Figure 1: Snapshot of a simulation showing three robots in the experimental arena. The dashed lines indicate the reference frame used in the experiments.

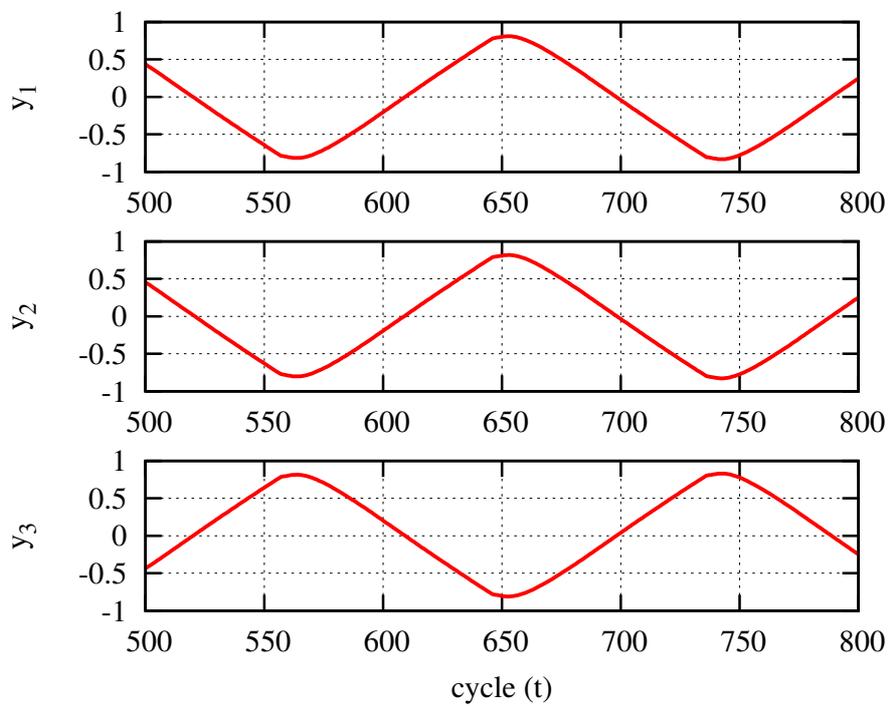


Figure 2: An example of synchronised motion of three robots. The y position of the robots is plotted against time.

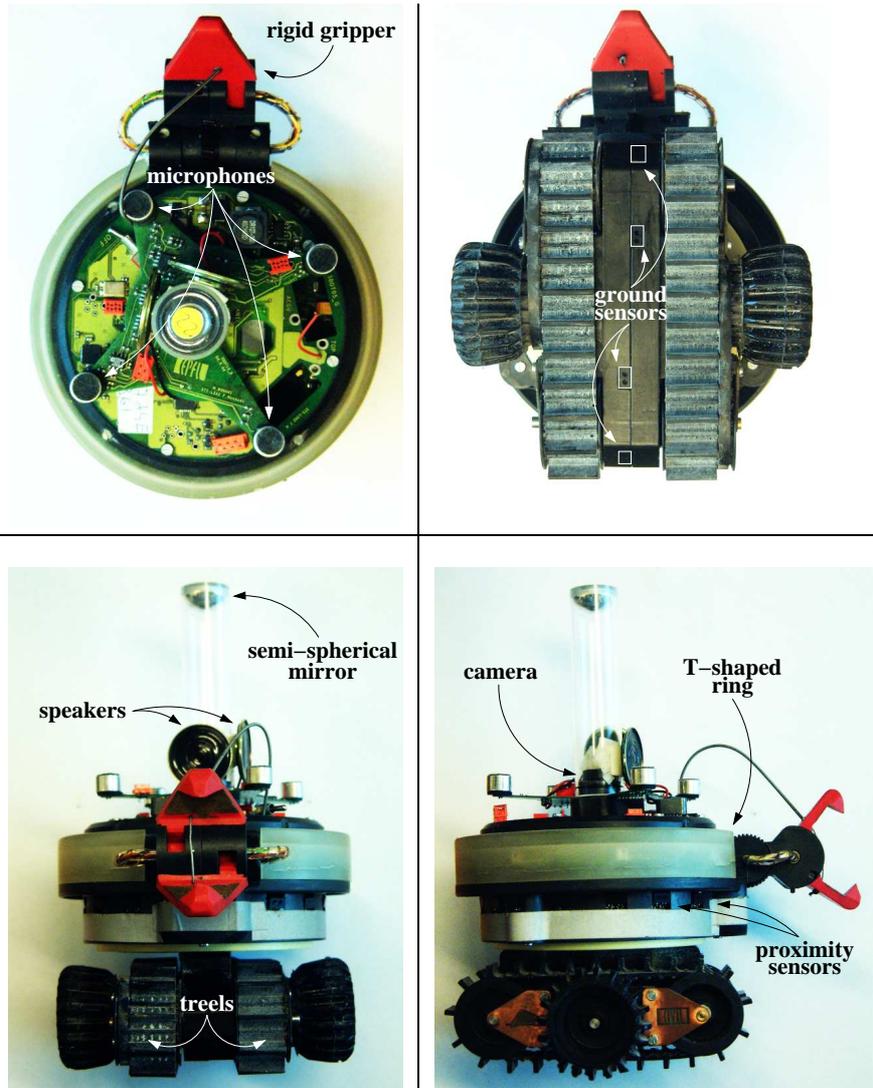


Figure 3: View of the *s-bot* from different sides. The main components are indicated (see text for more detail).

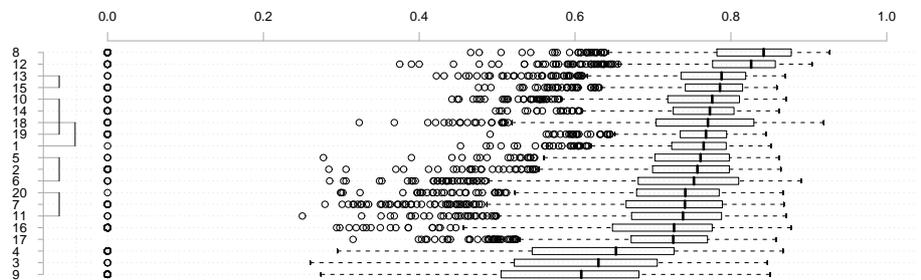


Figure 4: Post-evaluation results of the best evolved controllers c_i in each evolutionary experiment $i = 1, \dots, 20$. The performance is represented on the horizontal axis, and the controller number on the vertical axis. The boxplot displays the whole dataset: each box represents the inter-quartile range of the data, while the black vertical line inside the box marks the median value. The whiskers extend to the most extreme data points within 1.5 times the inter-quartile range from the box. The empty circles mark the outliers. Data from different controllers are sorted according to the median value. Moreover, statistical similarities are represented as vertical bars spanning over the controller numbers (see text for detail).

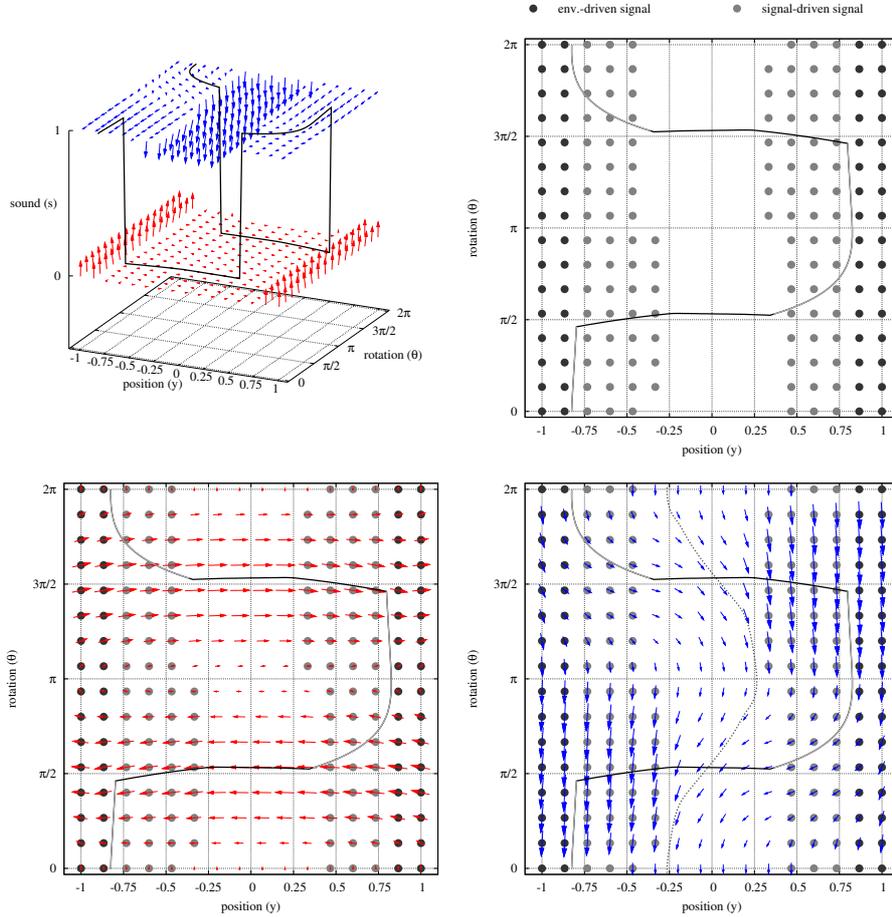


Figure 5: Individual behaviour produced by controller c_8 . Top-Left: 3D vector field showing the direction of variation and its magnitude for each point in the phase space. The θ dimension is characterised by 2π -periodic boundary conditions. The continuous line represents the limit cycle attractor. Top-Right: signalling behaviour of the controller for each position and orientation (see text for detail). The continuous line represents a projection of the limit cycle on the $y\theta$ plane: a black line colour indicates that the trajectory belongs to plane $s = 0$, while the grey colour corresponds to the portion of trajectory that belongs to plane $s = 1$. Bottom-Left/Right: projection on the $y\theta$ plane of the vector fields for a perceived signal $s = 0$ and $s = 1$. The dotted line in the bottom-right vector field represents the limit cycle for a continuous perceived signal forced to 1, despite the individual behaviour.

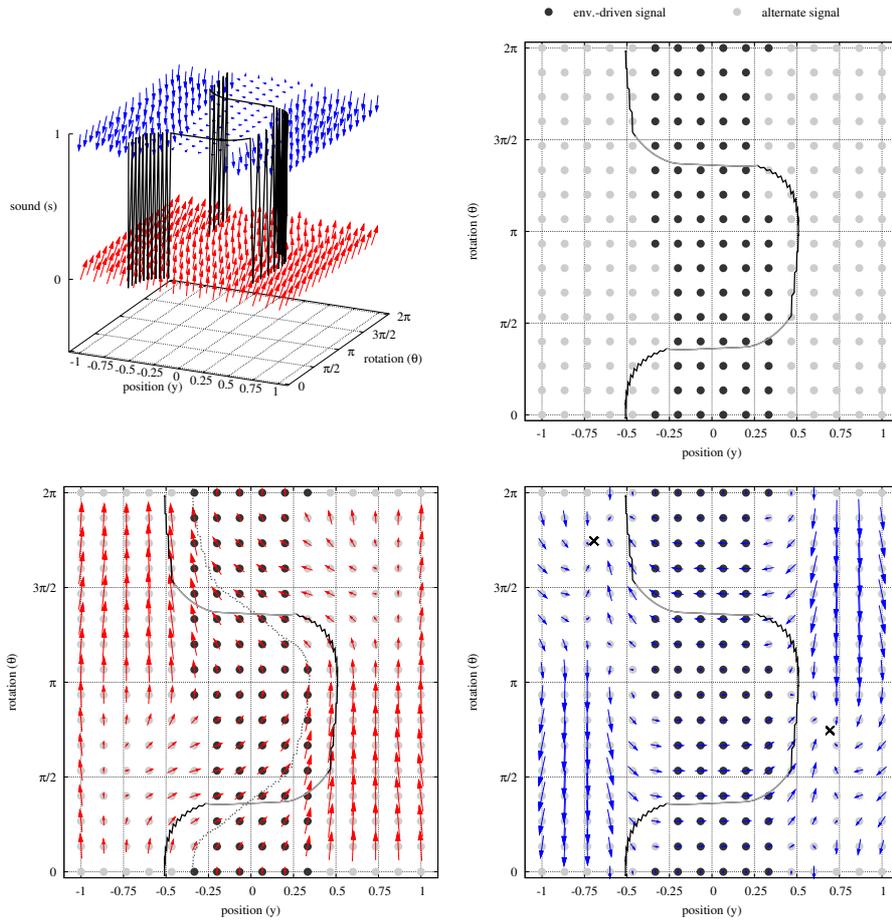


Figure 6: Individual behaviour produced by controller c_{10} . See the caption of Figure 5 for detail. Notice that the bottom-left plot is a vector field given by the average between the vector fields for $s = 0$ and $s = 1$, as it represents the average direction of movement during alternate signalling. In the bottom-left plot, the dotted line indicates the limit cycle attractor for a perceived signal forced alternate. In the bottom-right plot, the crosses indicate the fixed point attractors for a continuous perceived signal forced to 1, despite the individual behaviour.

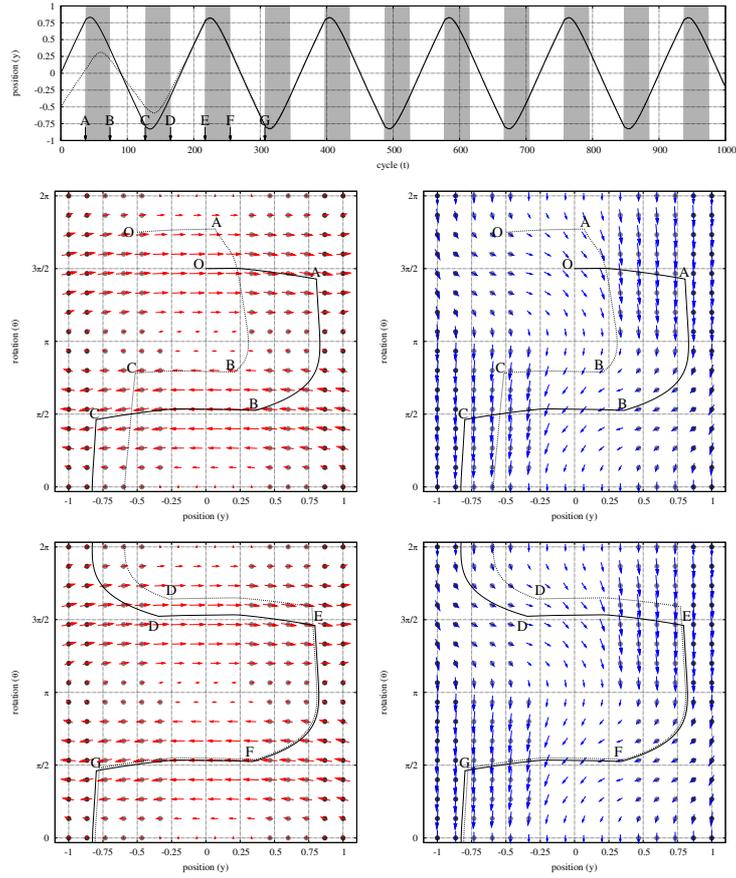


Figure 7: Synchronisation behaviour of controller c_8 . Top: the position y of two s -bots that synchronise is plotted through time. The grey bands in the background indicate that a signal is being perceived. Centre and bottom: vector fields for the conditions $s = 0$ (left) and $s = 1$ (right). For each point, the individual signalling behaviour is displayed as a dot with varying grey level (see also Figure 5). The trajectories of the two synchronising robots are shown, and relevant events are marked with capital letters. The same letters indicate the time of the corresponding events in the top graph.

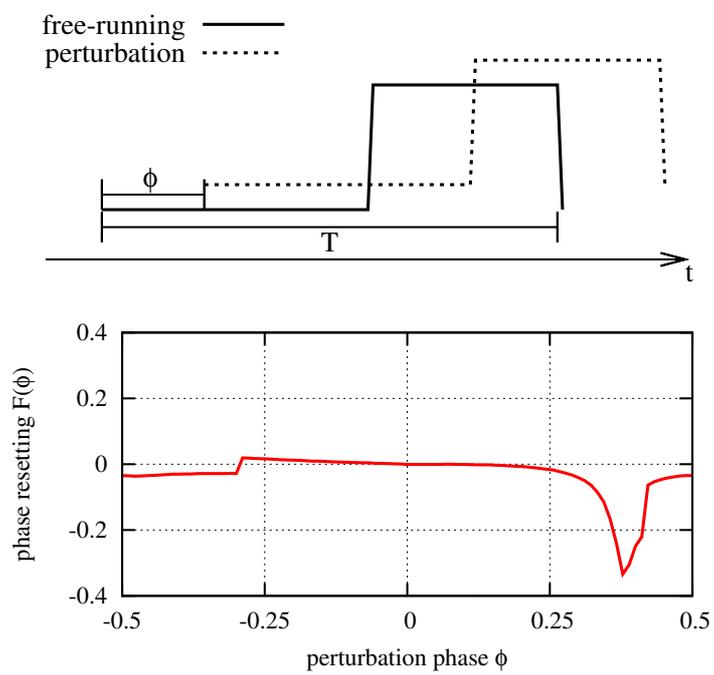


Figure 8: Top: Definition of the perturbation phase ϕ from the time delay between the signal of the free-running oscillator and the perturbation signal. Positive values correspond to a delayed signal, while negative values correspond to an anticipated signal. Bottom: Normalised phase response curve (PRC) for controller c_8 .

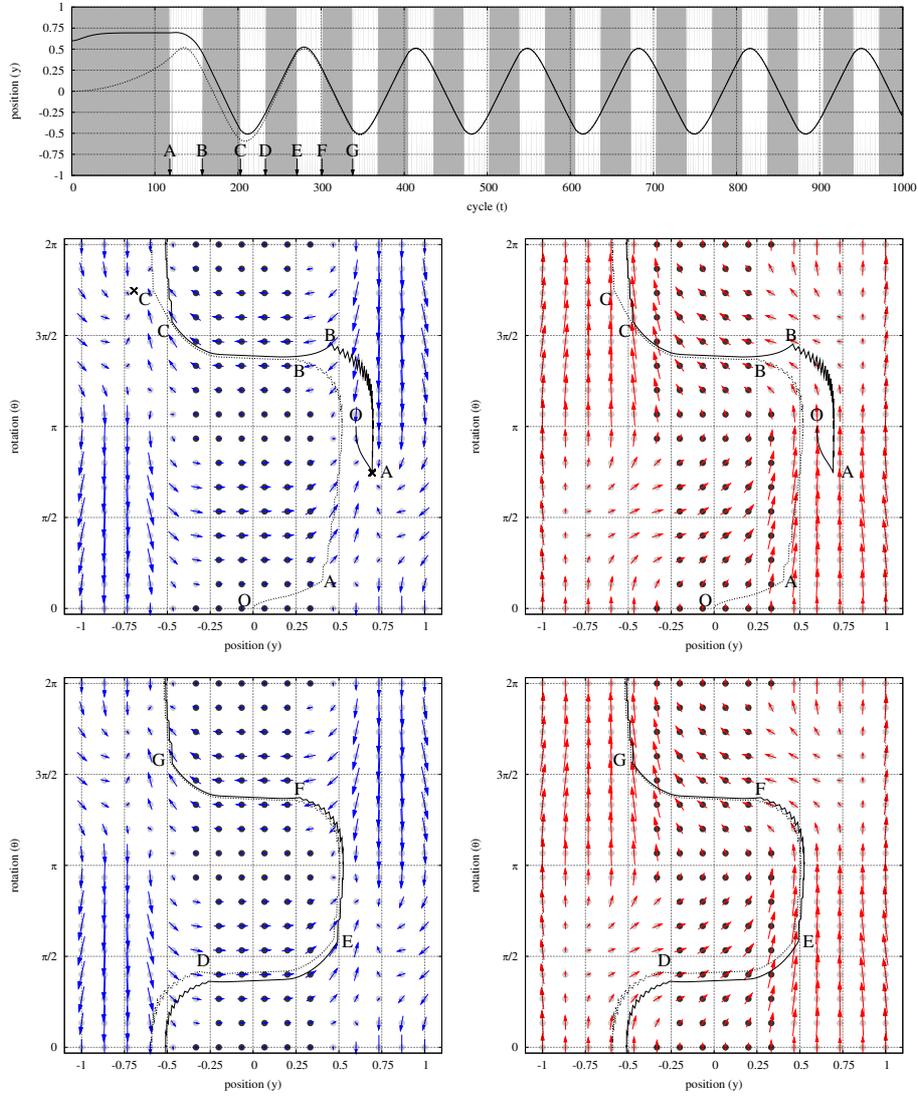


Figure 9: Synchronisation behaviour of controller c_{10} . Top: the position y of two s -bots that synchronise is plotted through time. The solid grey bands in the background indicate that a continuous signal is being perceived. The striped grey bands indicate an alternate signal. Centre and bottom: vector fields for the continuous signalling (left) and alternate signalling (right, see also Figure 6). For each point, the individual signalling behaviour is displayed as a filled circle with varying grey level. The trajectories of the two synchronising robots are shown, and relevant events are marked with capital letters. The same letters indicate the time of the corresponding events in the top graph.

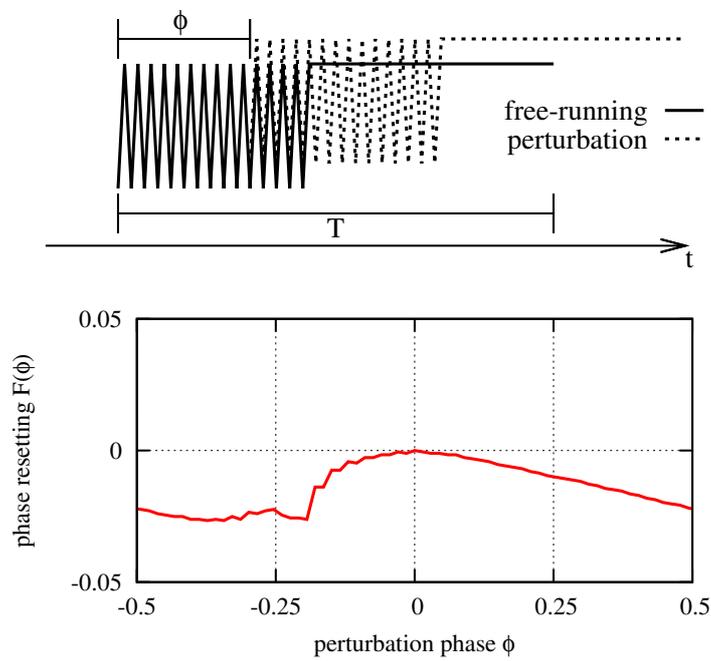


Figure 10: Top: Definition of the perturbation phase ϕ from the time delay between the signal of the free-running oscillator and the perturbation signal. Bottom: Phase response curve for the controller c_{10} .

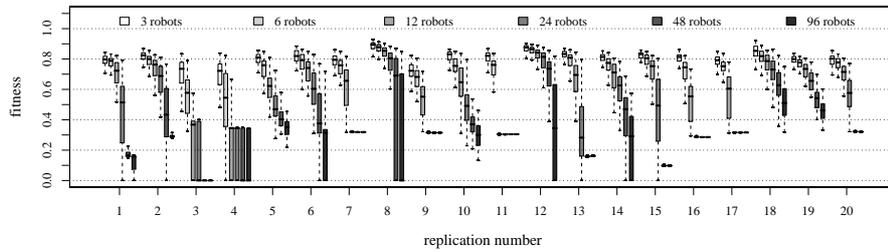


Figure 11: Scalability analysis. The boxplot shows, for each evolved controller, the performance obtained in tests with 3, 6, 12, 24, 48, and 96 *s-bots*. Each box represents the inter-quartile range of the data, while the black horizontal line inside the box marks the median value. The whiskers extend to the most extreme data points within 1.5 times the inter-quartile range from the box. Outliers are not shown.

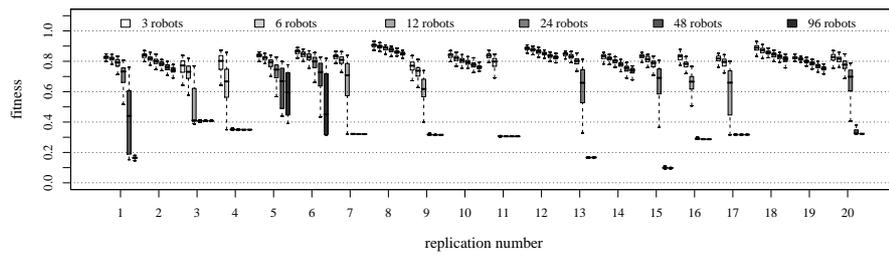


Figure 12: Scalability of the synchronisation mechanism. See the caption of Figure 11 for detail.

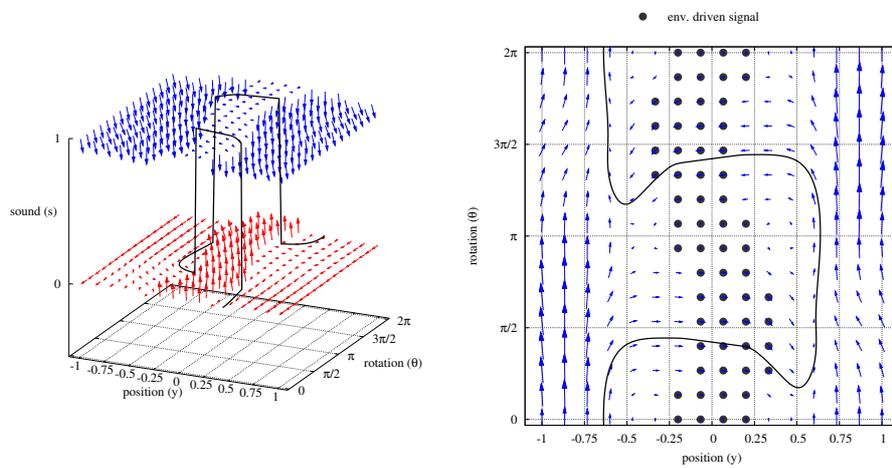


Figure 13: Individual behaviour of controller c_{13} . Left: the 3D vector field shows for each point in the phase space the direction of variation and its magnitude. Right: projection on the $y\theta$ plane of the vector field for a constant perceived signal $s = 1$. The black line represent the limit cycle for this condition. The black dots represent the *non-interaction area*.

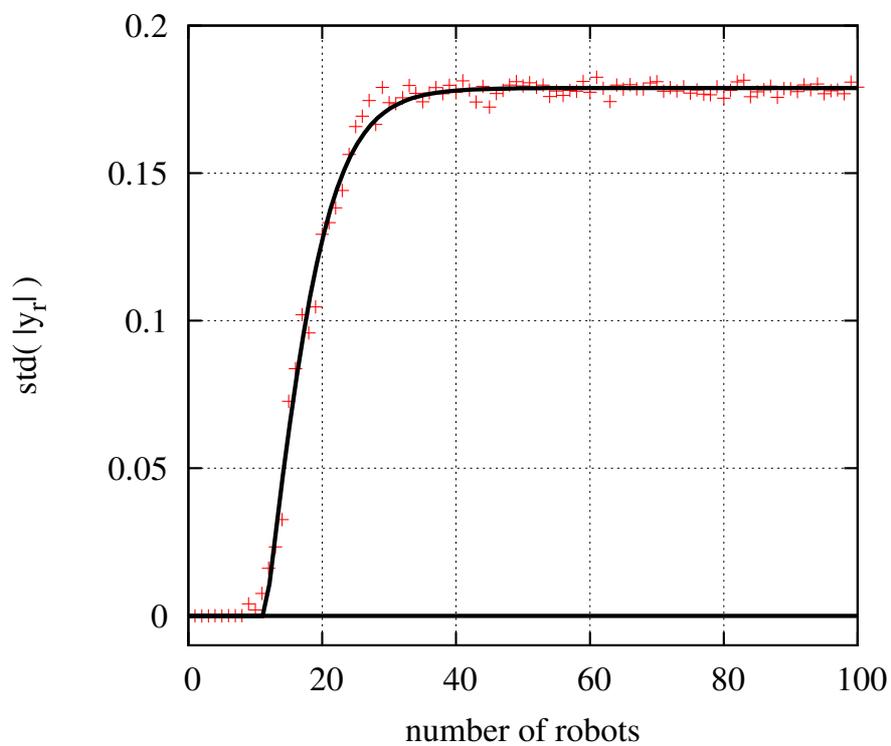


Figure 14: Average standard deviation of the absolute value of the y_r position for each s -bot r , plotted varying the group size R from 1 to 100. The black line represents a curve that fits the experimental data.

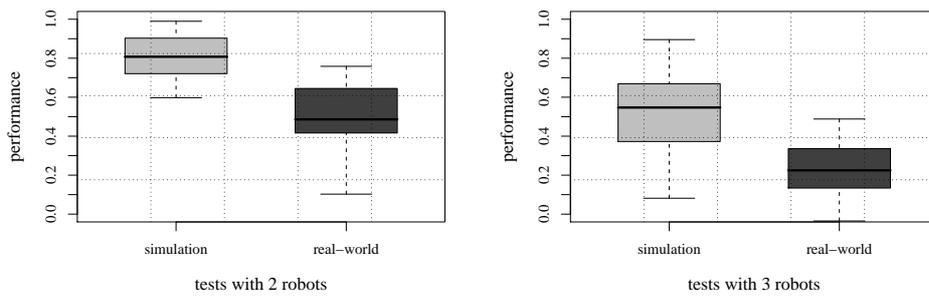


Figure 15: Comparison between the performance obtained in simulation and with physical robots.