A Robotic Ecosystem with Evolvable Minds and Bodies

Berend Weel, Emanuele Crosato, Jacqueline Heinerman, Evert Haasdijk, A.E. Eiben
VU University Amsterdam, The Netherlands
Email: b.weel@vu.nl, crosato.emanuele@gmail.com, {j.v.heinerman|e.haasdijk|a.e.eiben}@vu.nl

***Abstract*—This paper presents a proof of concept demonstration of a novel evolutionary robotic system where robots can self-reproduce. We construct and investigate a strongly embodied evolutionary system, where not only the controllers, but also the morphologies undergo evolution in an on-line fashion. Forced by the lack of available hardware we build this system in simulation. However, we use a high quality simulator (Webots) and an existing hardware platform (Roombots) which makes the system, in principle, constructible. Our system can be perceived as an Artificial Life habitat, where robots with evolvable bodies and minds live in an arena and actively induce an evolutionary process 'from within', without a central evolutionary agency or a user-defined synthetic fitness function.**

***Index Terms*—Embodied Evolution, Robotic Organisms, Evolutionary Robotics, Robotics, Robot Morphology**

I. Background, Motivation, Assumptions

This work forms a stepping stone towards the grand vision of strongly embodied evolution or the Evolution of Things as outlined in [1], [2]. The essence of this grand vision is to construct physical systems of animate artefacts that undergo evolution 'in the wild'. The approach behind this paper is based on using robots composed of simple mechatronic modules. However, the specific substrate is not very important for the concept itself and in general, the bodies can be made of traditional mechatronic components, (self-)assembled from simple modular units, formed by some soft material, 3D printed plastics, some fancy new stuff invented by material scientists, or any combination of these. The two key challenges we address here are

- A suitable reproduction mechanism for robot organisms. This is the core feature at the heart of the Evolution of Things.
- The integration of all evolutionary components into one system. This ensures that we obtain a full process with many successive generations.

An important aspect is that the robots are able to actively induce an evolutionary process 'from within' –without a central evolutionary agency– in real time and real space.

There are several reasons to be interested in self-reproducing robots. From the engineering perspective, the technology of evolvable robots offers possible applications in the future, where adapting the robot design and/or producing new robots during the operational period without human intervention is important. This can be the case in inaccessible environments, for example, colonies of mining robots that work in extreme depths under the surface of the Earth for extended periods, planetary missions, such as terraforming, deep sea explorations, or medical nano-robots acting as 'personal virus scanners' inside the human body.

Evolution can also be put to work in closer to home scenarios. Including the human in the loop to influence selection, the classic approach to designing and manufacturing robots can be changed into a modus operandi very much like breeding livestock. This can combine the human guidance (user selection) with the creative exploratory power of evolution as used today in *in silico* evolutionary design [3].

From the science perspective, the ecosystem with evolvable minds and bodies offers unprecedented opportunities for fundamental as well as applied research in embodied AI and ALife, because it eliminates the restriction of working with fixed morphologies. This opens the possibility to study the mind-body problem in a new way [4], [5], [6]. Paraphrasing Pfeifer and Bongard, [7], one could say that with the new technology we cannot only study how the body shapes the mind, but also how the mind shapes the body. There are also benefits for biological research where robots can be used as the substrate to create physical, rather than digital, models of biological systems and to study biological phenomena [8], [9].

In this paper we address two main challenges for the Evolution of Things: the creation of tangible physical artefacts with the ability to reproduce and the integration of all components of the Triangle of Life framework [10] into one working ecosystem. Forced by the lack of available hardware we build this system in simulation. However, we use a high quality simulator (Webots) and an existing hardware platform (Roombots) which makes the system, in principle, constructible. The use of a simulator for strongly embodied evolutionary robotics seems contradictory, but we deem it justified on the short term as a proxy until the hardware for self-reproducing robots becomes mature. Also on the long term, simulations can play an important role in the workflow, because they offer a practicable solution for system calibration and quick exploration of the design space without the high costs of using real hardware.

This paper presents a complete –albeit virtual– implementation of an ecosystem of self-reproducing robots with all components in place. Our proof of concept implements objective-free evolution along the lines of the mEDEA algorithm, which implicitly promotes robot movement through the arena [11]. The core of the evolutionary process is autonomous self-reproduction without any explicit objective: robots procreate by exchanging genetic material whenever they are in close proximity to one another (meeting = mating).

The most important research objective of this paper is to show that the system enables sustainable populations of evolving organisms that are born, learn and procreate autonomously. In contrast to 'regular' evolutionary algorithms, the size of the population in an ecosystem such as we envisage is not a parameter that the experimenter sets, but rather an observable: the robots interact (in our case, come close to one another) to mate autonomously. If, for whatever reason, there is not enough interaction, the population dies out. In other words, robots must procreate to maintain the population.

The second research objective is to investigate whether the robots, endowed with reinforcement learning capabilities, learn to locomote efficiently during their lifetime and how this learning ability evolves.

II. Related Work

A considerable body of Evolutionary Robotics research uses regular Evolutionary Algorithms to optimise controllers for robots with fixed morphologies [12], [13], [14].

To date there are no robotic systems where robots physically reproduce and create children with variation and heredity. There are systems with reproducing artificial creatures in simulation following the approach of [15], [7], but these are not physical, positioned in simple artificial environments, address only one function, e.g. walking, and typically far from being constructible in real life. More importantly, they do not form an open-ended ecosystem, but run along the lines of a traditional evolutionary algorithm with central selection: evolution is off-line and optimises a crisp user-defined objective. The number of systems with self-replicating hardware units is very low and they all miss some crucial aspects. The self-replicating machines of [16] and [17] produce exact clones of themselves without variation and heredity and hence they are not evolutionary. Furthermore, they are one trick ponies: the only thing their controllers are supposed to do is self-replication; they are not capable of operating in the environment and perform some task. The Evolvable Physical Self-Replicators of [18] lack controllers entirely: they are inanimate dumb artefacts in 2D. A few systems address co-evolving robot morphologies and controllers, but they circumvent the challenge of physical reproduction by performing evolution in simulation and constructing (some of) the evolved robots afterwards [19], [20]. The SYMBRION project realised modular robotic organisms where morphologies were reconfigurable, but not evolvable [21]. The autonomous robot modules could aggregate into organisms to negotiate some environmental obstacle and disassemble afterwards. However, such aggregated robot organisms were transient constructs and their reproduction was not an objective. In fact, it was not even possible because the system lacked an inheritable genetic code representing the design of an organism. Finally, there do also exist studies on 'embodied evolution', where evolution takes place in a physical robot population [22], [11]. However, the robots have a fixed morphology and they cannot produce children, evolution is limited to controllers inside the given bodies.

Whatever its shape, every robot requires some controller to be able to do anything. Therefore, there has been little research into evolving body shapes alone without regarding the controller. Meng et al. evolved a target shape for a self-reconfigurable modular robot for locomotion in a corridor, as well as climbing a stairs, and stepping over an obstacle [23]. Gross et al. evolved polymers of modules in a primordial 'soup' of modules of different types, the resulting organisms needed to gather energy to reproduce [24].

The co-evolution of body and mind was first investigated by Sims in his seminal paper [15], and his work was later built upon co-evolving morphologies and controllers in substrates ranging from lego to tensegrities [25] and materials of varying elasticity (soft robotics) [26]. Most research in this vein has been conducted with modular robots, e.g. in [19], [27], [28], [29]. All these approaches where mind and body evolve together are at heart a centralised evolutionary algorithm: the robots are instantiated in a simulator and their performance at some task (typically locomotion) is assessed and used as the basis for selection.

Such centrally orchestrated evolution contrasts with the vision set out above: it employs evolution as a force for optimisation. Research into artificial ecosystems is, in that sense, more in line with our vision of autonomously and asynchronously reproducing entities. Systems for the simulation of such artificial ecosystems abound in artificial life research, for instance Tierra [30] and Avida [31], two systems in which self-replicating machine code evolves by means of natural selection. Echo [32] is a simulation tool developed to investigate mechanisms that regulate diversity and information processing in systems comprised of many interacting adaptive agents. Systems such as SugarScape [33], Polyworld [34] and EcoSim [35] are used to study the evolution of agent control strategies through natural selection. None of these artificial ecosystems allow for the evolution of morphology – in fact, in many cases there is no morphology and agents are dimensionless.

Thus, while there is substantial related work on particular aspects of the system we envisage, there is to our knowledge no system that implements a complete artificial ecology with physically reproducing robots, where robot morphology and controllers co-evolve in an open-ended process induced by autonomous and asynchronous mating in the robots' environment.

III. System Description

The system is implemented[1] in the Webots [36] simulator. It is meant to be (A) an ecosystem (B) with reproducible robot organisms (C) which is based on the Triangle of Life framework.

[1]The code for our implementation is available through two projects on google code:
https://code.google.com/p/tol-project/
https://code.google.com/p/tol-controllers/

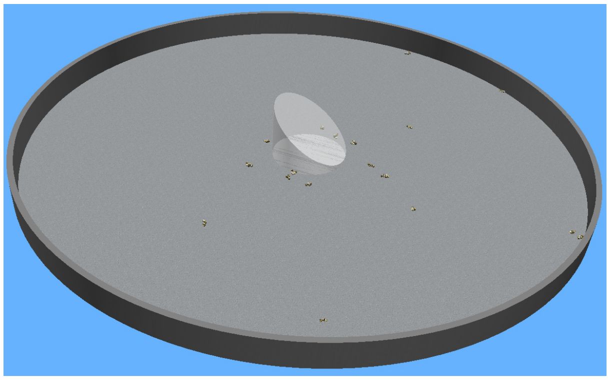


Figure 1: A screenshot of the whole environment. Mature and infant organisms coexist in a circular arena with a radius of 15m. The sliced cylinder in the centre is the birth clinic.

A. The Ecosystem design

Our ecosystem is a 3D world containing a fixed number of robotic modules (cf. Figure 1). All organisms are created using these modules and live inside a bounded circular arena. Infant organisms, still learning to move, as well as mature organisms, competing for reproduction, coexist in this arena. The environment is flat and void of obstacles, so organisms can move freely. A central facility, called birth clinic, is placed in the middle of the arena. The birth clinic fabricates the organisms and is where every new life cycle begins. Please note that even though organisms are constructed centrally the evolutionary system is decentralised: mate-/parent selection is done by the organisms and not centrally orchestrated. This is a fundamental difference with centralised evolution, as in our case reproduction is based on local interactions and with limited information.

B. Reproducible robot organisms

1) The Phenotype: The system simulates existing robotic modules called Roombots [37]. Every module consists of two cubic-like blocks, ten active connection mechanisms (ACMs) and three actuated joints (cf. Figure 2). Two of the three joints are located on each block's diagonal. They allow the two halves of a block to rotate relative to each other. The third one is located between the two blocks. All the joints are powered by step motors and are designed for continuous rotation. Each joint is provided with slip rings for electric power and

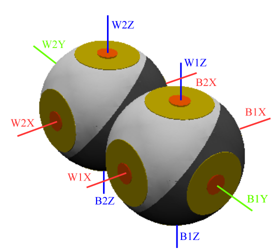
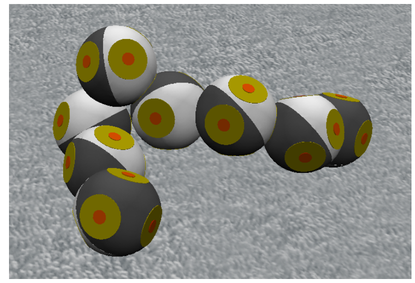
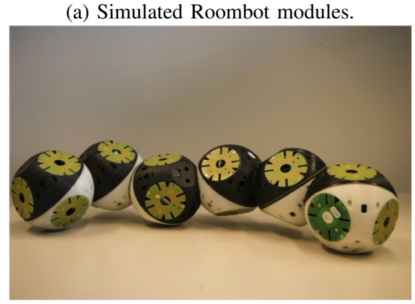


Figure 2: A Roombot module. The two blocks rotate on the middle joint and each of their halves rotate on the diagonals. Each Roombot features ten ACMs.



(a) Simulated Roombot modules.



(b) Real Roombot modules

Figure 3: Example of a robotic organisms

information transfer. Roombots can also communicate with each other and with other devices via bluetooth.

Each organisms consist of two or more Roombot modules attached using the ACMs, each module runs its own controller. The movement of such an organism is therefore an emergent property of the motion of their modules. The movement of all modules is governed by a special controller, the mind of the organism, which is run on a single module in each organism which therefore plays a special role. We call this module the *root* of the organism. The root has two extra functions with respect to normal modules. First, it drives the reinforcement learning process. Second, it provides an interface for the whole organism. In particular, the root module carries the organism's genome and governs the communication with other organisms and with the environment. Example organisms consisting of several Roombot modules can be seen in Figure 3, Figure 3b shows an example organism using real Roombot modules.

2) The Genotype: In this system the genome is a blueprint for an organism. It consists of an integer number d and a CPPN (Compositional Pattern-Producing Network) C that always has exactly two input nodes and one output node. The pair $< d, C >$ is called *body genome*, and it can be translated into a 2D build plan. Despite the build plan being only 2D, the built organism will take 3D shapes while moving.

Along with the body genome mentioned above, a *mind genome* is used. The mind genome defines the starting point for the learning algorithm used during lifetime learning. The mind genome depends on the learning algorithm used and in our case is a Policy for the RL PoWER algorithm explained in the next section. The combination of body and mind genomes will be referred as just *genome* from now on.

C. The Triangle of Life

The Triangle of Life is a framework for Artificial Life [10] that specifies the three principal life phases of most biological organisms: birth, infancy and mature life (cf. Figure 4).

A life cycle (triangle) starts with the *Birth process*, in which available modules are assembled into a new organism starting from a blueprint (morphogenesis). The second phase is *Infancy*, in which organisms improve their locomotion abilities before becoming fertile. The last phase of the triangle is the *Mature life*. In this phase organisms compete for resources and reproduction against other organisms. If fit enough, an organism will succeed in reproducing and it will generate a new genome. With the conception of a new genome a new triangle of life starts, without ending the previous one.

1) Birth Process:

a) The Birth Clinic: As mentioned all organisms are constructed using a birth clinic in the centre of the arena. Using a birth clinic gives two main advantages over a distributed method of construction. First, all free modules, including the ones of dead organisms, can be collected in a single location and become available for building new organisms. Second, evolution can be switched off by disabling the facility as a failsafe mechanism by denying the production of organisms.

We have placed a single birth facility in the middle of the arena, which has the shape of a sliced cylinder, this facility builds organisms based on a build plan that the organisms send during reproduction. Morphogenesis takes place high above the cylinder. The newborn organism is released from this position and slides down the diagonal section of the cylinder, until it reaches the floor of the arena. The cylinder rotates before the creation of each organism so that new organisms are distributed in all directions around the facility. When an organism is created, the necessary modules are moved from the storage to the building location, in the position specified by the build plan. An autonomous collection mechanism of the available modules is an important aspect of the implementation, however in this version of the system we assume that this mechanism can be engineered and use a shortcut.

b) Morphogenesis: During morphogenesis the genome of the organisms is translated using the following procedure. (a) A virtual square grid of size d is created. Each grid cell represents a possible position for a single block of a Roombot. The coordinates of each cell are serially fed as input to C to get a real number as output. The cells with a value lower than a certain threshold are excluded and they will never host any block.

(b) Among the remaining cells, the one with the highest value is chosen. A second cell is similarly selected among the neighbouring cells. This pair of cells will host the first module.

(c) Another module is added by choosing a pair of cells as in step (b), but searching only among the neighbouring cells of the already placed modules. An additional constraint denies the modules from forming cycles.

Step (c) is repeated until no other couples of cells can be identified for hosting a module. The first module added to the build plan is assumed to be the root node. A build plan generated as described can be empty or made of only one module, in either case the plan is considered not valid and no actual organism is built. The materialisation of a building plan is completely logical, meaning that no physical process assembles the modules. An autonomous assembling procedure is important for a hardware ecosystem, however, this feature is not yet designed in this version of the system. In a hardware scenario, this operation can be performed by an ad-hoc assembly device.

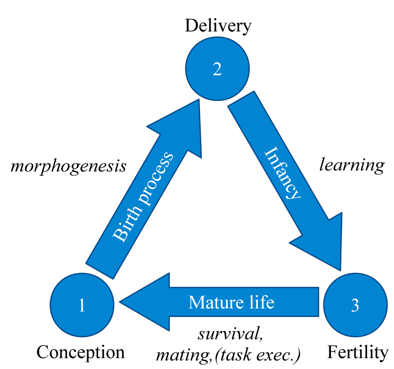


Figure 4: The Triangle of Life. Three pivotal moments span the triangle: 1) Conception: A new genome is activated, construction of a new organism starts. 2) Delivery: Construction of the new organism is completed. 3) Fertility: The organism becomes ready to conceive offspring.

2) Infancy: The infancy is an important phase during an organisms' lifetime. First the learning of the organism is initiated in this phase and allows the organism to reach a certain level of competence before being subjected to selection mechanisms. Moreover the infancy phase is an important test phase, if the organism does not reach a certain minimum performance it can be decided to not allow the organism to enter the reproductive pool at all.

Organisms are not required to perform any specific task and are free to move in any direction. This reduces the locomotion problem to gait learning. A non-trivial problem in itself. Particularly, it requires the generation of rhythmic functions for the activation of the organisms' step motors. The RL PoWER algorithm has been chosen for gait learning in this project based on previous investigations [38]. Learning is not restricted to the infancy period, but organisms continue learning for their full lifetime.

The initial policy, defined by the mind genome, is periodically modified and evaluated by RL PoWER, for a fixed number of times. A policy is a set of parametrised cyclic splines, one for each joint of a Roombot module, and each of them describes the servo motor angle as a function of time. A spline is defined by a set of n control points. Each control point is defined by (t_i, α_i), where t_i represents time and α_i the angle. $t_i \in [0, 1]$ is defined as

$$t_i = \frac{i}{n-i}, \forall i = 0, ..., (n-1)$$

and $\alpha_i \in [0, 1]$ is freely defined, except that the last value is enforced to be equal to the first. These control points are used for cyclic spline interpolation.

The RL PoWER implementation follows the description by Jens Kober and Jan Peters [39] and [38]. If the organism is from the initial population the algorithm starts by creating the initial policy π_0 with as many splines as there are motors in the organism. These splines are initialised with n values of 0.5 and then adding Gaussian noise. Otherwise the minds of the parents are combined as explained later and this mind is used as the initial policy. The initial policy is then evaluated after which it is adapted. This adapted controller is evaluated and adapted again until the stopping condition is reached. Adaptation is done in two steps which are always applied: exploitation and exploration. In the exploitation step, the current splines $\hat{\alpha}$ are optimised based on the outcome of previous controllers, this generates a new set of splines.

$$\hat{\alpha}_{i+1} = \hat{\alpha}_i + \frac{\sum_{j=1}^{k} \hat{\Delta}\alpha_{i,j} R_j}{\sum_{j=1}^{k} R_j}$$

where $\hat{\Delta}\alpha_{i,j}$ represents the difference between the parameters of the i-th policy and j-th policy belonging to a ranking of the best k policies seen so far and R_j its reward. In the exploration phase policies are adapted by applying Gaussian perturbation to the newly generated policy.

$$\hat{\alpha}'_{i+1} = \hat{\alpha}_{i+1} + \hat{\varepsilon}_{i+1}, \hat{\varepsilon}_{i+1} \sim \mathcal{N}(0, \sigma^2)$$

where $\hat{\alpha}_{i+1}$ are the parameters after the exploitation step, $\hat{\alpha}'_{i+1}$ the parameters after the exploration step and $\hat{\varepsilon}_{i+1}$ values drawn from a Gaussian distribution with mean 0 and variance σ^2.

Each controller is evaluated for a fixed time as follows:

$$R_i = \left(100 \frac{\sqrt{\Delta_x^2 + \Delta_y^2}}{\Delta_t}\right)^6$$

where Δ_x and Δ_y is the displacement over the x and y axes measured in meters and Δ_t the time of an evaluation.

3) Reproduction: During mature life organisms periodically spread messages containing their genome in a limited range. Organisms within this range receive the message and store the genome. After a certain interval, an organism reproduces by recombining its own genome with a received one, which is selected randomly. It is important to notice that organisms that have not entered the range of any other organism cannot reproduce. Based on this interaction we assume that the faster an organism, the higher the probability for it to receive and send genomes and therefore reproduce, this constitutes a distributed implicit parent selection mechanism. The choice of an implicit selection, that uses no fitness function, is important, as it makes the system closer to a real ecosystem. Moreover, in complex systems formulating a definition of being fit might be hard or even impossible. This might not sound so essential for an environment as simple as this, however, additional elements such as tasks or obstacles can be added to dramatically increase the complexity.

A design problem arises with this selection mechanism. Since all organisms begin their lives next to the birth clinic, slow organisms will still be close to the centre of the arena after becoming mature. This also means that they will be close to each other, which in turn leads to them procreating with high probability To counter this scenario, an additional constraint is introduced: even if mature, an organism cannot reproduce and neither spread its genome before crossing a virtual circular boundary placed at a certain distance from the birth clinic.

A new genome is always created by applying crossover on the parents' genomes and mutating the result. The grid size of the body genome is the average of the parents' values which is then mutated by random noise drawn from a normal distribution. This result is then rounded to the nearest integer. The CPPN variation operators are taken from Stanley and Miikkulainen [40], recurring connections are disallowed. A new policy is created with as many splines as the smallest policy of the parents, each spline containing as many points as the shortest of the parents. Every point of the new policy is copied from either parent with equal probability. The policy is also perturbed with random noise drawn from a normal distribution. The newly created genome is then sent to the birth clinic, where morphogenesis of the new organism takes place and a new life starts. The birth clinic is assumed to be reachable from any position in the arena.

No death selection mechanism has been implemented in place of survivor selection, instead each organism has a fixed time to live. This decision mimics a limited energy supply given to each organism at the beginning of life.

D. Parameters & Initialisation

The following list describes the main parameters of our system and the value used in our experiments.

Environment

`Total run length` — **`36000 s`**
The total length of a run in seconds.

`Number of modules` — **`45`**
The total number of Roombots in the system.

`Arena size` — **`15 m`**
The radius of the circular arena.

`Fertility boundary` — **`5 m`**
Radius of the boundary to cross to become fertile.

RL PoWER

`Learning duration` — **`2000 s`**
Total duration of the infancy phase.

`Number of evaluations` — **`200`**
Total number of policies evaluations.

`Standard deviation` — **`0.008`**
The initial standard deviation.

`Standard deviation decay` — **`0.98`**
The standard deviation decay factor.

`Ranking size` — **`10`**
Number of best policies to compare with.

`Start parameters` — **`2`**
Starting number of parameters to define a spline.

`End parameters` — **`20`**
End number of parameters to define a spline.

Genome

`Grid starting size` `3`
Size of the grid at the beginning of evolution.

`Grid minimum size` `3`
Minimum size assigned to the grid.

`Size mutation rate` `0.6`
Chance of mutating the grid size.

`Size mutation strength` `1.5`
Magnitude of the grid size mutation.

`Mind mutation rate` `0.5`
Chance of mutating a single point of the policy.

`Mind mutation strength` `1`
Magnitude of the mutation of a point of the policy.

`Threshold` `0`
Cell's threshold for hosting a Roombot block.

Organism

`Organisms emitter range` `5`
Action range of the organisms' emitter.

`Mating interval` `500 s`
Interval after which organisms try to reproduce.

`Spreading genome interval` `3 s`
Interval after which organisms spread the genome.

`Time to live` `8000 s`
Total life duration including the infancy.

Finally we initialised the population by creating a random body plan and mind for the first organism. Then, for as long as a successful mating has not yet occurred, another organism is initiated between 60 and 180 seconds after the previous one.

IV. First Runs and Findings

We ran the system 25 times with the parameters as described in the previous section. The first aspect we investigate is the dynamics of the population.

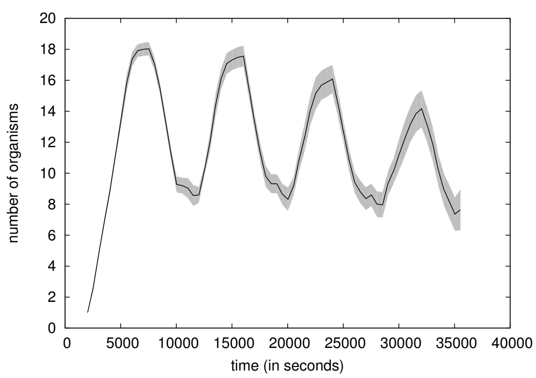


Figure 5: Number of mature organisms over time. The black line shows the number of mature organisms averaged over all runs with a 95% confidence interval in grey.

The black line in Figure 5 shows the mean number of mature organisms over time, averaged over all runs with a 95% confidence interval. The initial population, starting with one organism, becomes mature at time step 2000 and as long as there are available modules, the population size increases. Hereafter a cyclic pattern arrises because the system has to wait until organisms die (after 8000 time steps) and wait until the new born organisms will be mature. Longer runs are necessary to see whether the population size stabilizes. But during the length of the run, the population will not die out and has a minimum size of 8.

After looking into the population dynamics we considered reproduction and inspected the number of mating actions.

We divide the runs into 18 intervals of 2000 seconds each, based on the length of their mature life. In Figure 6, we show the number of unique genomes received (i.e. of different organisms encountered) by a mature organism before each reproduction. Each box plot represents data over all mature organisms in all runs where reproduction took place in that time slot. We can see that over almost the entire run, the median number of received genomes is alternating between 1 and 2. This alternating behaviour seems to follow the same cyclic pattern as in Figure 5. Logically, when there are more mature organisms in the arena, it is more likely to collect more genomes.

The next aspect we investigate is the learning behaviour. Figure 7 shows the learning performance at the start of the infancy and at the end of its lifetime for the first 10 organisms and the last 10 organisms aggregated over all runs. The first box plot shows the performance of the first 30 evaluations of the first 10 organisms of each run. The second box plot shows the first 30 evaluations of the last 10 organisms of each run. Similarly the third and fourth box plots show the performance of the last 30 evaluations of the first and last 10 organisms.

First we notice that learning does take place: the median performance of organisms at the end of their lifetime is much higher than at the beginning of their lifetime. This is in line with the previous work on our learning algorithm RL PoWER [38]. A new aspect we introduced in this system is the evolution of the mind, to analyse the effect of this evolution

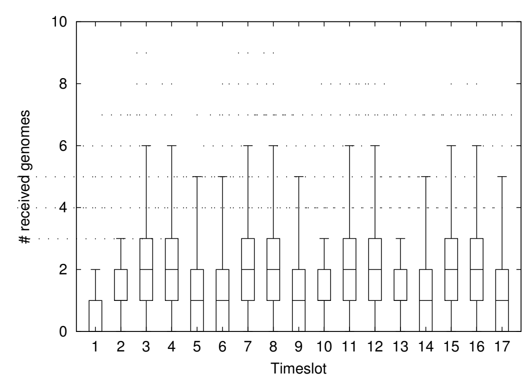


Figure 6: Received genomes at reproduction. The x axis represents the time of the total run split in 18 time slots.

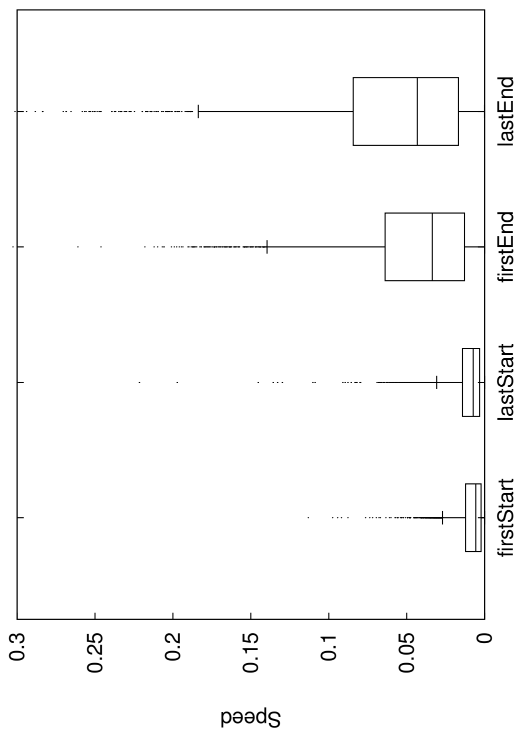


Figure 7: Learning performance of organisms over all runs. The firstStart box plot shows the performance of the first 30 evaluations of the first 10 organisms of each run. The lastStart box plot shows the first 30 evaluations of the last 10 organisms of each run. Similarly the firstEnd box plot and the lastEnd box plot show the performance of the last 30 evaluations of the first and last 10 organisms

we looked at the difference in performance between early and late organisms. The idea is to check whether the initial mind evolves towards a policy that leads to a better learning during the organisms lifetime. We can see the starting performance of the first 10 organisms and last 10 organisms is very similar, later organisms do not seem to have a head start in their learning. Similarly there is little difference in the end speeds of the organisms, showing that the learning algorithm is capable of learning in a similar fashion with the evolved shapes.

Last, but not least, we investigated reproduction from a spatial perspective. Intuitively, one would expect that faster organisms reproduce more because they encounter more would-be partners. (Recall that the evolutionary system does not specifically select for faster organisms; meeting is mating, regardless of particular properties of the robots.) We therefore examine the speed of the organisms and the number of children they have had.

Figure 8 displays the related data with the speed of the parent shown plotted on the y-axis for each number of children. We can see that the speeds show high variation, especially for small numbers of children (0 to 4). To gain further insights into reproduction we also examined where mating takes place in the arena.

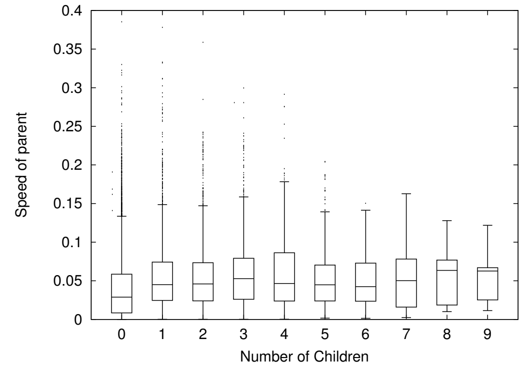


Figure 8: Relation between a robots' speed and its number of children. Every datapoint consists the total produced children so far, which corresponds to the box plot number, and the latest speed evaluation of the parent.

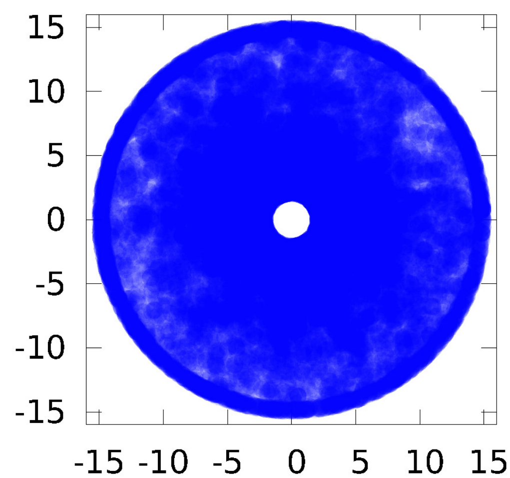


Figure 9: Mating locations over all runs. The circle corresponds with the arena size. The higher the density in the graph, the more mating takes place.

The map in Figure 9 shows the mating locations over all runs, each transparent blue circle represents a single mating occurrence, the darker a certain location, the more matings took place there. We can clearly see the contours of the arena; the white circle in the middle is the birth clinic. We can see that the density is high everywhere, but highest near the birth clinic and at the perimeter of the arena. This means that organisms move away from the birth clinic.

V. Conclusion and Further Research

This paper presented a proof of concept demonstration of a novel evolutionary robotic system where robots can self-reproduce. In particular, we a) specified multi-modular Roombot structures as the robotic phenotypes, b) designed appropriate encoding for their morphologies and controllers as genotypes, and c) implemented variation operators to produce new offspring genotypes from parent genotypes.

With this reproduction mechanism at the core, we have built a robotic ecosystem where robots live and evolve in their (simulated) environment driven by autonomous and asynchronous mating, without a central evolutionary agency and a user-defined synthetic fitness function. The resulting system is the first one we know of that implements a complete artificial ecology in which robot morphology as well as controllers can evolve in an on-line fashion driven by the environment.

Our first stated research objective was "to show that the system enables sustainable populations of evolving organisms that are born, learn and procreate autonomously." We have indeed shown that, in the environment as we defined it, with movement through the environment promoting fecundity, the robots evolve to move around the arena and so encounter mates. This allows them to procreate, resulting in a viable population that spans several generations.

The second research objective was "to investigate if robots, endowed with reinforcement learning capabilities, learn to locomote efficiently during their lifetime and how this learning ability evolves." The robots are indeed capable of lifetime learning and profoundly improve their locomotion capabilities over their lifetime. The evolution of the initial minds does not, however, seem to have a great impact on the learning ability, but this may change with longer evolutionary runs.

We intend to use our system for multiple avenues of research, including the emergence of embodied intelligence, the co-evolution of body and mind under different circumstances and physical evolution itself in an ecosystem. The choice for implementing the system in simulation is a temporary solution until feasible technology for reproducing robots becomes available. The use of an existing hardware platform (Roombots) makes our system, in principle, constructible.

References

[1] A. E. Eiben, S. Kernbach, and E. Haasdijk, "Embodied artificial evolution," *Evolutionary Intelligence*, vol. 5, no. 4, pp. 261–272, 2012.

[2] A. Eiben, "In Vivo Veritas: towards the Evolution of Things," in *Parallel Problem Solving from Nature – PPSN XIII*, ser. LNCS, T. Bartz-Beielstein, J. Branke, B. Filipič, and J. Smith, Eds., vol. 8672. Springer, 2014, pp. 24–39.

[3] P. Bentley and D. Corne, *Creative Evolutionary Systems*. Morgan Kaufmann, San Francisco, 2002.

[4] M. Anderson, "Embodied cognition: A field guide," *Artificial Intelligence*, no. 149, pp. 91–130, 2003.

[5] M. Hoffmann and R. Pfeifer, "The implications of embodiment for behavior and cognition: animal and robotic case studies," in *The Implications of Embodiment: Cognition and Communication*, W. Tschacher and C. Bergomi, Eds. Imprint Academic, 2012, pp. 31–58.

[6] F. Iida, R. Pfeifer, L. Steels, and Y. Kuniyoshi, Eds., *Embodied Artificial Intelligence*, ser. LNAI. Springer, 2004, no. 3139.

[7] R. Pfeifer and J. Bongard, *How the Body Shapes the Way We Think*. MIT Press, 2006.

[8] D. Floreano and L. Keller, "Evolution of adaptive behaviour in robots by means of darwinian selection," *PLOS Biology*, vol. 8, no. 1, p. e1000292, 2010.

[9] J. Long, *Darwin's Devices: What Evolving Robots Can Teach Us About the History of Life and the Future of Technology*. Basic Books, 2012.

[10] A. E. Eiben, N. Bredeche, M. Hoogendoorn, J. Stradner, J. Timmis, A. Tyrrell, and A. Winfield, "The triangle of life: Evolving robots in real-time and real-space," in *Advances in Artificial Life, (ECAL) 2013*, P. Lió, O. Miglino, G. Nicosia, S. Nolfi, and M. Pavone, Eds. MIT Press, 2013, pp. 1056–1063.

[11] N. Bredeche, J.-M. Montanier, W. Liu, and A. F. Winfield, "Environment-driven distributed evolutionary adaptation in a population of autonomous robotic agents," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 18, no. 1, pp. 101–129, 2012.

[12] S. Nolfi and D. Floreano, *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, 2000.

[13] J. Bongard, "Evolutionary robotics," *Communications of the ACM*, vol. 56, no. 8, pp. 74–85, 2013.

[14] D. Floreano, P. Husbands, and S. Nolfi, "Evolutionary robotics," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, vol. Part G.61, pp. 1423–1451.

[15] K. Sims, "Evolving 3d morphology and behavior by competition," *Artificial Life IV*, pp. 28–39, 1994.

[16] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson, "Self-reproducing machines," *Nature*, vol. 435, no. 7039, pp. 163–164, 2005.

[17] V. Zykov, E. Mytilinaios, M. Desnoyer, and H. Lipson, "Evolved and designed self-reproducing modular robotics," *Robotics, IEEE Transactions on*, vol. 23, no. 2, pp. 308–319, 2007.

[18] N. Virgo, C. Fernando, B. Bigge, and P. Husbands, "Evolvable physical self-replicators," *Artif. Life*, vol. 18, no. 2, pp. 129–142, Apr. 2012. [Online]. Available: http://dx.doi.org/10.1162/artl_a_00056

[19] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, no. 406, pp. 974–978, 2000.

[20] H. Lund, "Co-evolving control and morphology with lego robots," *In Morpho-functional Machines: The New Species*, pp. 59–79, 2003.

[21] P. Levi and S. Kernbach, Eds., *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*, ser. Cognitive Systems Monographs. Springer, Berlin, Heidelberg, New York, 2010, vol. 7.

[22] R. A. Watson, S. G. Ficici, and J. B. Pollack, "Embodied evolution: Distributing an evolutionary algorithm in a population of robots," *Robotics and Autonomous Systems*, vol. 39, pp. 1–18, 2002.

[23] Y. Meng, Y. Zhang, and Y. Jin, "Autonomous self-reconfiguration of modular robots by evolving a hierarchical mechanochemical modeling a hierarchical mechanochemical model," *Computational Intelligence Magazine*, vol. 6, no. 1, pp. 43–54, 2011.

[24] R. Gross, S. Magnenat, L. Küchler, V. Massaras, M. Bonani, and F. Mondada, "Towards an autonomous evolution of non-biological physical organisms." in *Advances in Artificial Life. Darwin Meets von Neumann.*, 2011, pp. 173–180.

[25] M. Khazanov, B. Humphreys, W. Keat, and J. Rieffel, "Exploiting dynamical complexity in a physical tensegrity robot to achieve locomotion," in *Advances in Artificial Life, ECAL XII*, P. Liò, O. Miglino, G. Nicosia, S. Nolfi, and M. Pavone, Eds., 2013, pp. 965–972.

[26] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, "Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding," in *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO '13. New York, NY, USA: ACM, 2013, pp. 167–174. [Online]. Available: http://doi.acm.org/10.1145/2463372.2463404

[27] G. Hornby, H. Lipson, and J. Pollack, "Generative representations for the automated design of modular physical robots," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 4, pp. 703–719, 2003.

[28] J. Bongard, "Morphological change in machines accelerates the evolution of robust behavior," *Proceedings of the National Academy of Sciences*, vol. 108, no. 4, pp. 1234–1239, 2011. [Online]. Available: http://www.pnas.org/content/108/4/1234.abstract

[29] M. Komosinski and J. Polak, "Evolving free-form stick ski jumpers and their neural control systems," in *Proceedings of the National Conference on Evolutionary Computation and Global Optimization*, Poland, 2009, pp. 103–110.

[30] T. Ray, "Is it alive or is it ga?" *Proceedings of International Conference on Genetic Algorithms*, 1991.

[31] C. Adami and B. T. C, "Evolutionary learning in the 2d artificial life system avida," *Proceedings of Artificial Life IV*, pp. 377–381, 1994.

[32] J. H. Holland, "Echoing emergence: Objectives, rough definitions, and speculations for echo-class models," in *Complexity*, G. A. e. a. Cowan, Ed. Cambridge, MA, USA: Perseus Books, 1999, pp. 309–342.

[33] J. Epstein and R. Axtell, *Growing Artificial Societies: Social Sciences from Bottom Up*. Brooking Institution Press and The MIT Press, 1996.

[34] L. S. Yaeger, V. Griffith, and O. Sporns, "Passive and driven trends in the evolution of complexity," in *Proceedings of Artificial Life XII: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*, S. e. a. Bullock, Ed. Cambridge: MIT Press, 2008.

[35] R. Gras, W. A, and A. Aspinall, "An individual-based evolving predator-prey ecosystem simulation using a fuzzy cognitive map model of behavior," *Artificial Life*, vol. 15, no. 4, pp. 423–463, 2009.

[36] Webots, "http://www.cyberbotics.com," commercial Mobile Robot Simulation Software. [Online]. Available: http://www.cyberbotics.com

[37] A. Sproewitz, A. Billard, P. Dillenbourg, and A. J. Ijspeert, "Roombots-mechanical design of self-reconfiguring modular robots for adaptive furniture," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 4259–4264.

[38] M. D'Angelo, B. Weel, and A. Eiben, "Hyperneat versus rl power for online gait learning in modular robots," in *Proceedings of EvoApplications 2014: Applications of Evolutionary Computation*, 2014.

[39] J. Kober and J. Peters, "Learning motor primitives for robotics," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 2112–2118.

[40] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary computation*, vol. 10, no. 2, pp. 99–127, 2002.