

Indirectly Encoded Sodarace for Artificial Life

In: *Proceedings of the 12th European Conference on Artificial Life (ECAL-2013, Taormina, Italy)*. Cambridge, MA: MIT Press

Paul Szerlip and Kenneth O. Stanley

Department of EECS, University of Central Florida, Orlando, FL 32826-2363

pszerlip@eecs.ucf.edu

Abstract

The aim of this paper is to introduce a lightweight two-dimensional domain for evolving diverse and interesting artificial creatures. The hope is that this domain will fill a need for such an easily-accessible option for researchers who wish to focus more on the evolutionary dynamics of artificial life scenarios than on building simulators and creature encodings. The proposed domain is inspired by Sodarace, a construction set for two-dimensional creatures made of masses and springs. However, unlike the original Sodarace, the *indirectly encoded Sodarace* (IESoR) system introduced in this paper allows evolution to discover a wide range of complex and regular ambulating creature morphologies by encoding them with *compositional pattern producing networks* (CPPNs), which are an established indirect encoding originally introduced for encoding large-scale neural networks. The result, demonstrated through a technique called novelty search with local competition (which are combined through multiobjective search), is that IESoR can discover a wide breadth of interesting and functional creatures, suggesting its potential utility for future experiments in artificial life.

Introduction

An important aim of artificial life is to uncover the conditions that yield interesting discoveries in evolutionary domains. For example, researchers studying *open-ended evolution* (Channon, 2001a,b; Maley, 1999; Ray, 1992; Standish, 2003; Yaeger, 1994) seek to produce dynamics that yield a continual stream of novel and potentially more complex phenotypes. Other approaches to evolving lifelike creatures focus less on the evolutionary dynamics than on a particular property like morphology (Joachimczak and Wrobel, 2012), locomotion (Clune et al., 2011; Lehman and Stanley, 2011a), or both (Auerbach and Bongard, 2012; Bongard and Paul, 2000; Hornby and Pollack, 2002; Krcak, 2007; Lehman and Stanley, 2011b; Sims, 1994). The promise of such investigations is that they can potentially reveal key conditions that lead to the most compelling or natural results.

However, a significant obstacle to entering this research area is the lack of standardized artificial creature domains and genetic encodings. Indeed, in almost all such work

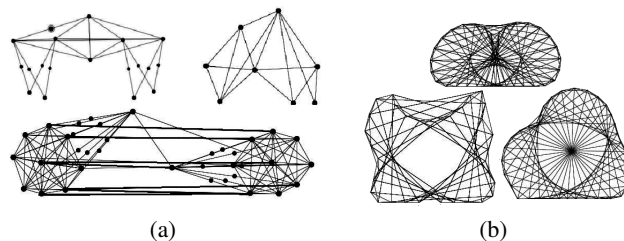


Figure 1: **Sodarace Examples.** Human-designed racers (a) exhibit diverse strategies and morphologies for ambulation while those produced through the evolutionary optimizer (b) share an amoeba-like morphology and similar ambulation.

researchers design their own domains and encodings from scratch, creating a high barrier to entry. In effect there is no out of the box domain that is easy to integrate quickly into a larger experimental framework. For example, an experiment aiming to research the impact of different evolutionary dynamics or selection pressures on open-ended discovery presently requires not only formulating a hypothesis, but also an entire domain and genetic encoding, which may not be the main motivation for the investigation in the first place. While of course sometimes researchers will prefer to build such experiments from the ground up, the availability of an easy, lightweight option that makes it possible to focus quickly on broader evolutionary questions would nevertheless be beneficial to the field overall.

The aim of this paper is to highlight such a lightweight option for a broad creature-space with low barrier to entry. The key concept is to introduce an encoding that opens a *Sodarace-like domain* of two-dimensional ambulatory creatures (McOwan and Burton, 2013, 2005) to broad and diverse evolutionary exploration and discovery. Sodarace is a simulation engine for two-dimensional creatures made of masses, springs, and muscles that ambulate based on the construction of their body morphology. It was originally introduced with human designers in mind, allowing them to construct their own Sodaracers by hand and then race them competitively, yielding a collection of diverse and interest-

ing human-designed morphologies reminiscent of the output of a successful artificial life world (figure 1a). In fact, one version of Sodarace even included an evolutionary optimizer, but because of the simplicity of its genetic encoding, the evolved morphologies only represent a small corner of the possibilities suggested by the human designs (figure 1b).

To open up such a domain to more interesting evolution, a Sodarace-inspired domain and encoding called *indirectly encoded SodaRace* (IESoR) is introduced in this paper that is designed explicitly for evolutionary exploration. In particular, the possibility of evolving a range of natural yet diverse morphologies exhibiting regularities across their structure is created by *compositional pattern producing networks* (CPPNs) (Stanley, 2007) evolved by the HyperNEAT algorithm (Gauci and Stanley, 2010; Stanley et al., 2009). While HyperNEAT and CPPNs were originally introduced to evolve large-scale neural networks, because the creatures in IESoR are also defined fundamentally by nodes and connections, HyperNEAT can in effect also evolve creature bodies with the same regularities and symmetries seen in CPPN-encoded neural networks (Clune et al., 2011).

With this new Sodarace-inspired implementation and an established indirect encoding behind it, the potential for the system to evolve both far more variety and quality than the original Sodarace evolver is demonstrated through a *novelty search with local competition* (Lehman and Stanley, 2011b), which is a recent method for efficiently surveying the range of possibilities that exist within a particular design space. The main outcome is that IESoR indeed introduces a rich and easily accessible platform for exploring a wide variety of interesting creatures with low simulation cost and concrete visual payoff.

Background

This section contains an overview of Sodarace and MINS, the online projects that serve as the inspiration for IESoR, as well as a brief review of the NEAT and HyperNEAT methods used to encode creature morphology in IESoR.

Sodarace

The Sodarace project is a simple two-dimensional physics world consisting entirely of masses, springs, and basic oscillatory muscles (McOwan and Burton, 2013, 2005). The goal in Sodarace is to create virtual robots and race them in different environments. Both the robots and the environments are usually hand-crafted by users. However, to aid in creating robots, a construction kit is provided to allow discovery and exploration by the community (McOwan and Burton, 2013, 2005).

The Sodarace project was originally conceived as a type of online Olympics meant to test humans against machine intelligence at the task of designing robot racers. In fact, one redesign of the software includes an evolutionary algorithm that optimizes morphologies for racing. Reflecting the

software’s educational aspirations, an online repository of creatures and all relevant software packages are accessible in a centralized location (McOwan and Burton, 2013). At the peak of popularity, Sodaconstructor, the tool for creating the creatures, was played by about a million active users (McOwan and Burton, 2005), suggesting its potential as a platform for exploration and discovery.

MINS

While Sodarace was a beacon for user creativity, the project itself was created over a decade ago and the community has declined since then. Nevertheless, the peak popularity of the project suggests the domain has wide appeal, though some aspects of the original software make Sodarace inaccessible to academic research. For example, as a closed source Java applet, certain parameters of the races cannot be modified because the Sodaconstructor user interface does not provide user access.

To address the obstacle to research, Stefan Westen created Mins Is Not Sodarace (MINS), an open source replica of the Sodarace environment (Westen, 2013). MINS is fully compatible with Sodarace, allowing the user to import design and environments from Sodaconstructor. By creating an alternative open-source environment, MINS allows adjusting parameters that are hardcoded inside the Sodarace domain. In an effort to curb cheating by Sodarace creatures, MINS alters the environment ceiling size, starting velocities, and maximum movement speeds.

MINS is an inspiration for the work in this paper in part because the variety of creature types found with the Sodaconstructor suggests the space of creatures is rich. MINS also shows that replicating the Sodarace environment is feasible and lightweight, while maintaining backwards compatibility. The primary principle extracted from Sodarace and MINS is the use of masses, springs, and muscles to construct varied creatures inside a customizable physics environment (Westen, 2013).

NEAT and HyperNEAT

Evolving morphology and control is familiar to artificial life (Auerbach and Bongard, 2012; Bongard and Paul, 2000; Hornby and Pollack, 2002; Krcak, 2007; Lehman and Stanley, 2011b; Sims, 1994). In this spirit, recent additions to Sodarace include a utility with an evolutionary algorithm for creating and optimizing racers inside of the software (McOwan and Burton, 2005). In particular, inside of the Sodarace “Kiosk,” users are presented with a limited interface for designing a creature. The user can adjust the number of virtual nodes and muscles along with the amplitude and frequency at which the muscles oscillate. The evolutionary algorithm then searches constrained by those parameters through selection and mutation to find the fastest racer possible. Inside of the Sodarace Kiosk, by default creatures appear to be circular in nature with criss-crossed inner connec-

tions. With gravity, the resulting creatures take on the shape of a semi-circular blob (figure 1b). The Sodarace community refers to this creature morphology as being “amoeba-like.” There is a Sodarace utility called the Amoebamatic that aids users in constructing these amoeba racers.

However, the most interesting handcrafted creatures from Sodarace generally do not exhibit amoeba-like properties. That is, the included genetic encoding is highly constrained to a small subset of all the interesting possibilities. Ideally, the encoding and evolutionary algorithm for evolving such racers would be able to search a wide breadth of possible creatures, which would make this kind of domain relevant to artificial life. Yet to efficiently search such a space requires a principled encoding capable of searching variable levels of complexity.

The first step towards this end in IESoR is the NeuroEvolution of Augmenting Topologies algorithm (NEAT) (Stanley and Miikkulainen, 2002, 2004). Although NEAT was originally introduced as a method for evolving artificial neural networks (ANNs), a major appeal of NEAT is its ability to evolve increasingly complex structures of any type, so that evolutionary search is not limited to a fixed space of possibilities.

Conveniently for IESoR, a method called Hypercube-based NEAT (HyperNEAT) (Gauci and Stanley, 2010; Stanley et al., 2009) builds on NEAT to help it encode large connectivity patterns with natural regularities like symmetry and repetition of structure. While such regularities are useful for neural networks, they can also in principle benefit *bodies* made of connections and joints in a similar way.

The key ingredient behind HyperNEAT is an indirect encoding called a *compositional pattern producing network* (CPPN) (Stanley, 2007). The idea behind CPPNs is that geometric patterns can be encoded by a composition of functions that are chosen to represent common regularities. The internal structure of a CPPN is a weighted network, similar to an ANN, that denotes which functions are composed and in what order, which means that instead of evolving ANNs as it normally does, NEAT can evolve CPPNs that generate a connectivity pattern across a network.

The difference in this paper is that the pattern encoded by CPPNs is interpreted as a body plan for a Sodarace-like creature instead of a neural network. In fact, in a significantly different domain, Auerbach and Bongard (2012) encoded the bodies of three-dimensional ambulating creatures with CPPNs. The indirect CPPN encoding can compactly represent patterns with regularities such as symmetry, repetition, and repetition with variation (Secretan et al., 2011; Stanley, 2007), which are also exhibited by many natural organism morphologies on Earth. In fact, part of the inspiration for CPPNs derives from observations of natural bodies (Stanley, 2007).

To understand how a composition of functions could represent these regularities, simply by including a Gaussian

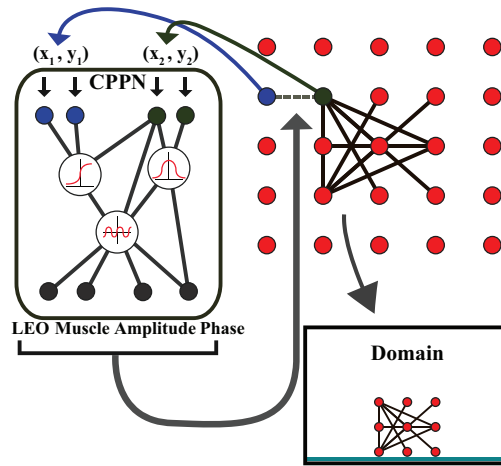


Figure 2: **Creating a Sodarace-like body using a HyperNEAT CPPN.** In regular HyperNEAT, the CPPN (left) would query the substrate (right) to determine the weights and presence (determined by the LEO output; Verbancsics and Stanley (2011)) of its connections. However, in IESoR the CPPN outputs the muscle, amplitude, and phase parameters for each queried connection instead of a connection weight. That way, the CPPN in effect describes the properties of a Sodarace body instead of a neural network, yet still with the same benefits of HyperNEAT as usual. The resultant creature is placed into a two-dimensional world where it attempts to ambulate.

function, which is symmetric, the output pattern of a CPPN can become symmetric. A periodic function such as sine creates segmentation through repetition. Most importantly, repetition with variation (e.g. such as the fingers of the human hand) is easily discovered by combining regular coordinate frames (e.g. sine and Gaussian) with irregular ones (e.g. the asymmetric x -axis). For example, a function that takes as input the sum of a symmetric function and an asymmetric function outputs a pattern with imperfect symmetry.

In this way, CPPNs produce regular patterns with subtle variations. The potential for CPPNs to represent patterns with motifs reminiscent of patterns in natural organisms has been demonstrated in several studies (Secretan et al., 2011; Stanley, 2007), and suggests such an encoding has potential in the domain of two-dimensional creatures.

Formally, CPPNs in HyperNEAT are functions of geometry (i.e. locations in space) that output connectivity patterns whose nodes are situated in n dimensions, where n is the number of dimensions in a Cartesian space. Consider a CPPN that takes four inputs labeled x_1 , y_1 , x_2 , and y_2 . This point in four-dimensional space also denotes the connection between the two-dimensional points (x_1, y_1) and (x_2, y_2) , and the output of the CPPN for that input thereby represents the weight of that connection (figure 2). By querying ev-

ery possible connection among a pre-chosen set of points in this manner, a CPPN can produce a connectivity pattern, wherein each queried point is a node position. Because the connections are produced by a function of their endpoints, the final structure is produced with knowledge of its geometry. In effect, the CPPN paints a pattern on the inside of a four-dimensional hypercube that is interpreted as the isomorphic connectivity pattern, which is the origin of the name hypercube-based NEAT (HyperNEAT). Connectivity patterns produced by a CPPN in this way are called substrates so that they can be verbally distinguished from the CPPN itself, which has its own topology. While the substrate in the original HyperNEAT is interpreted as an ANN, in IESoR the substrate is a creature's body, as explained next.

Approach

This section describes the implementation details of IESoR, and explains the variant of HyperNEAT that enables it to create and evolve body plans.

IESoR

IESoR implements three primary properties derived from Sodarace and MINS (figure 2):

1. The environment is two-dimensional and creatures consist solely of masses, springs, physical joints, and muscles.
2. In creature bodies, masses are implemented by nodes and springs are connections attached at the joints.
3. Muscles manipulate the length of connections, leading to motion.

In contrast to more complicated three-dimensional domains (Auerbach and Bongard, 2012; Krcah, 2007; Lehman and Stanley, 2011b; Sims, 1994), to support robust alife evolution IESoR is designed to be simple to modify and inexpensive to simulate. In the spirit of accessibility and extensibility of the Sodarace project, IESoR implements a Sodarace-like simulator in javascript built on top of Box2D (box2d.org), an open-source two-dimensional rigid body physics engine. There is a small performance hit for programming in a scripting language, but javascript allows the domain to be accessible through the browser for most modern computing devices, from phones to tablets to more traditional PCs. In addition, Box2D physics enables rich environments for testing creature morphologies. Finally, Box2D has been ported to most popular programming languages, which means IESoR could be ported without significant effort.

Encoding Morphologies with HyperNEAT

Bodies inside of IESoR consist of masses with variable or fixed length constraints. Each constraint, or connection, is represented by a distance joint in Box2D (i.e. a constraint on the length between two masses) and has three distinct properties:

1. The joint is either variable or fixed length (i.e. a muscle or a bone).
2. The change in distance during muscle contraction is the muscle amplitude.
3. The phase shift of the sinusoidal function controlling muscle length is the muscle phase.

Fixed length connections, or bones, do not receive a magnitude or phase from the CPPN.

Recall that HyperNEAT paints a four-dimensional pattern across the weights of a network by querying a CPPN for every pair of nodes in the substrate. The insight in this paper is to take this concept of a substrate and extend it to two-dimensional morphologies. Instead of painting a pattern of weights across the substrate, the CPPN encodes both what joint constraints should exist between masses on a two-dimensional plane and their three virtual properties (i.e. bone or muscle, amplitude, and phase). For this purpose, the CPPN requires four outputs (as shown in figure 2).

Before clarifying how a HyperNEAT substrate can be used to represent a morphology, it is important to consider the placement of bones and muscles in natural body plans. The skeletal system is crucial to mobility at a fundamental level. Equally important to where bones are placed in a body plan is the concept of where bones *are not* placed. If a rough representation of the human body was drawn on a small grid of dots, the principle of symmetry is as important as the fact that there is no bone connecting the tip of the foot to the top of the skull. Morphologies generated in IESoR ideally also should usually respect this simple principle of locality.

Conveniently for this purpose, HyperNEAT can be expanded with a special Link Expression Output (LEO) (Verbancsics and Stanley, 2011) to generate an *expression pattern* that controls whether connections are expressed at different locations independently of other CPPN outputs. In Verbancsics and Stanley (2011), HyperNEAT with LEO was seeded with a bias towards favoring locality although evolution could adjust this bias during search.

To generate a morphology using an n -by- n grid of nodes as the substrate, for each node location in the substrate (figure 2), the CPPN queries all other node positions. The (x, y) coordinate of nodes i and j are denoted as (x_i, y_i) and (x_j, y_j) , respectively. The input into the CPPN is thus x_i, y_i, x_j, y_j , and there are four outputs. First, the LEO output (which is a step function) is checked for a positive value. If LEO is positive, a connection is placed between nodes i and j from (x_i, y_i) to (x_j, y_j) . Then the output that determines whether the connection is a bone or a muscle is queried. If the output value is below a pre-defined *muscle threshold*, the connection becomes a fixed-length constraint. Otherwise, the constraint is a muscle, and the amplitude and phase of the muscle contraction are read from the remaining two CPPN outputs. Finally, to further reduce complexity in the resultant morphologies and keep computational costs

low, pairs of points greater than a third of the diagonal length of the substrate are not queried while constructing the two-dimensional creatures. An example of a fully constructed morphology is shown in the lower right of figure 2.

After assembling the masses and joints, the bodies are placed in a simple Box2D environment consisting of the ground, gravity, and friction. As the world is simulated, muscles oscillate according to the amplitude and phase values defined by the CPPN, while bones remain a fixed length. Creatures occupy distinct Box2D environments, and nodes cannot collide with each other.

Experiment

Though its creatures are mainly hand-crafted, Sodarace shows that the space of possible two-dimensional body types is likely filled with creatures capable of movement. As noted in Section 2.3, the Sodarace Kiosk went on to create an automated approach to generating creatures, but resulted in a highly restricted space of bodies. The experiment described in this section is designed to show that not only is an automated approach capable of designing two-dimensional walkers, but the method can also produce a *wide variety* of different means for locomotion, thereby giving hope for further application of Sodarace-like creatures in artificial life.

Novelty Search and Local Competition

To best demonstrate the morphological diversity possible in IESoR, Pareto multiobjective search (based on NSGA II) (Deb et al., 2002) including both novelty and local competition (Lehman and Stanley, 2011b) is implemented to explore the space of body types. Lehman and Stanley (2011b) first applied Pareto multi-objective search with novelty and local competition to yield a diverse group of ambulating three-dimensional morphologies all within a *single run* of evolution. Maintaining and exploiting diversity across evolution is both an impressive and important part of validating the potential for future artificial life research with IESoR.

The first of the three objectives that make up novelty search plus local competition is novelty search, which was introduced by Lehman and Stanley (2008, 2011a) to avoid the common pitfall of evolution prematurely converging on a deceptive objective. Novelty search aligns well with the aim of this experiment because the hope is to find a diversity of novel creatures. Joachimczak and Wrobel (2012) have shown before that novelty search can be effective for this purpose. The characterization of creature novelty for the novelty search component can significantly impact evolution and strongly bias the resulting creatures discovered. In this experiment, novelty search characterizes creatures by their width, height, and mass (as measured by the number of nodes and the sum of the connection lengths) at the first time-step of the simulation, which should lead to a visually diverse population. The novelty metric is the squared Euclidean distance separating two individuals in this charac-

terization space, and thus the novelty of a creature is proportional to how different its starting morphology is from that of other creatures currently in the population. Such a characterization space especially encourages creatures with varying widths, heights, and masses.

The second objective, local competition to walk farthest, forces individuals to compete *only* with those who are characterized as similar (Lehman and Stanley, 2011b). The idea is that within novelty search it is possible to push individuals who are similar with respect to the behavior characterization to compete locally to be the best of their type. That way, globally novelty search probes a wide variety of possibilities, but locally individuals optimize to be the best they can. In IESoR, creatures who are locally close share similar widths, heights, and masses, ideally indicating a similar morphology. Local competition is the mechanism for pressuring individuals with related morphologies towards more effective locomotion.

As in Lehman and Stanley (2011b), the Pareto multi-objective search has three objectives: novelty, local fitness, and finally *genotypic diversity*. The genotypic diversity objective encourages exploring innovative genotypes by assigning higher values to more novel genotypes. That way, new genotypes created by HyperNEAT are not initially penalized and thereby have a chance to optimize to reach their potential. This genetic diversity objective is in effect a multiobjective-compatible substitute for the usual speciation mechanism in NEAT, which serves the same purpose. Additionally, the genotypic diversity objective is also localized within the characterization space; similar in motivation to that of local competition, local genetic diversity ensures that genotypic diversity is not only exploited in those characterization niches in which such diversity is incidentally most easily expressed.

In all setups, the distribution of individuals in behavioral space as well as their overall performance is recorded. The idea is to quantify how much morphological diversity is discovered and maintained and how well each behavioral niche is being exploited overall throughout a run.

Experimental Parameters

The overarching multiobjective algorithm is based on NSGA II (Deb et al., 2002). The population size is 120, and a run consists of 1,200 generations, resulting in 144,000 total genomes evaluated. The nearest-neighbor size for novelty search and local competition is 20. The three morphology dimensions used to characterize novelty (i.e. width, height, and mass) are rescaled so that their values fill the range between zero and three. The selection method for NSGA II was tournament selection (with tournament size two), and other parameters followed precedent Lehman and Stanley (2011b), which in turn used the parameters of Krcak (2007).

Results

The intention of the experiment is to demonstrate that a wide variety of walkers exists in the encoding space defined by IESoR, thereby establishing the viability of IESoR for future alife research. Thus, as opposed to machine learning experiments aimed at demonstrating optimality, the aim in this experiment is to show both diversity and competence. Recall also that novelty search plus local competition is designed to return a significant coverage of possible solutions from a single long run. There is precedent for demonstrating the diversity that results from such a search. For example, Lehman and Stanley (2011b) measured the height and mass of three-dimensional morphologies from novelty search plus local competition to show the breadth of morphologies discovered by evolution, while Joachimczak and Wrobel (2012) used principal component analysis (PCA) to demonstrate coverage across morphological space after novelty search. Following this precedent, to quantify IESoR's ability to create diverse walkers, PCA is run across characterizations all 144,000 creatures from 1,200 generations of evolution to create a visualization of the resultant diversity.

In particular, to characterize morphological diversity in IESoR for the purpose of visualization, three dimensions that describe gross creature characteristics (i.e. width, height, and mass) are projected into a two-dimensional space by the PCA algorithm. However, while PCA with this information can reveal the diversity across the morphological space, the goal of this analysis is also to give a sense of the competence of such creatures as well. That way it becomes possible to observe the *diversity of competent creatures* instead of just diversity overall. Therefore, in the visualization of the PCA output in figure 3, to ensure the graph shows the diversity of only competent walkers, only points for walkers that ambulate beyond 200 units (which is several times a creature's maximum body length) are displayed. Furthermore, the size of each point's radius is proportional to the absolute distance traveled by the creature beyond 200 units.

Because thousands of points result, the visualization is further refined to reduce clutter and ensure that each point represents a genuinely unique individual. For this purpose, the plane is discretized into 40×40 equally sized "bins." Creatures are placed into bins according to the coordinate assigned by the PCA process. Conceptually, each bin thus represents a similar area in morphological space, and the creature assigned to a bin that traveled the farthest among all in that bin is chosen as the representative of that bin. That way, the circles in figure 3 show the best performance for the morphological class represented by its respective bin, and each circle represents a distinct morphological class. Any bin without a representative (shown as empty space in figure 3 lacks an individual who could ambulate at least 200 units.

Of the 1,600 possible bins, 450 are filled with individuals who can ambulate the minimum distance, covering in total 28.1% of all possible bins. Furthermore, the visualization

in figure 3 exhibits the breadth of coverage of competent morphologies. In effect, IESoR with novelty search plus local competition uncovered hundreds of unique and effective ambulation methods covering a significant breadth of conceivable strategies.

Equally important as this quantitative perspective is a qualitative analysis of the breadth of behaviors. It is important to note that every behavior in figure 3 can be viewed at <http://eplex.cs.ucf.edu/ecall13/demo/PCA.html> through a special online interface where the user can click on any point and see the corresponding creature behavior. This fast interactive visualization of hundreds of creatures is possible in part due to the lightweight, inexpensive nature of Sodarace-like creatures, which is one of their potential advantages for researchers in artificial life. Figure 3 also shows a sampling of morphologies, while figure 4 shows a subset of those at different stages of ambulation.

An additional important further qualitative observation is the significantly broader diversity seen in IESoR compared to the original Sodarace evolver's amoeba-like creatures shown in figure 1b. Among those that can be observed are gaits based on loping (degrading into pushing) (figures 3a/4a), pogo-stick hopping (3b/4b), multiple cascading octopus legs (3c/4c), dragging (3d/4d), complex galloping (3e), sliding and pumping (3f), and bouncing into a long dive (3g). Some strategies depend on an initial burst of propulsion, while others rely on stable and consistent ambulation. Some of the very best gaits (largest circles in figure 3) involve galloping or hopping, though even among the very best the diversity of approaches is significant.

Discussion

The most important implication of the results is that Sodarace-like domains do contain a diversity of viable walkers that can be systematically discovered with the right encoding and selective pressures. Unlike with the early evolver application built for Sodarace, morphologies evolved in IESoR do not exhibit only a single stereotypical organizational motif. Instead, they ambulate in many different ways, from legged-style locomotion to serpentine pulsation to periodic lunges, suggesting the potential for more elaborate applications of this kind of technology in the future.

For example, because the IESoR creatures are inexpensive to simulate, they are amenable to web-based interaction. In fact, the demonstration at the website actually simulates evolved discoveries in real-time through javascript, obviating the need even for video. This ease of simulation means that IESoR creatures can smoothly integrate into interactive evolutionary applications, artificial life worlds, or even interactive simulations allowing human intervention. Additionally, the framework can potentially be extended to three dimensional creatures using current browser technologies.

Investigations relevant to artificial life on open-ended evolution (Channon, 2001b; Maley, 1999; Ray, 1992;

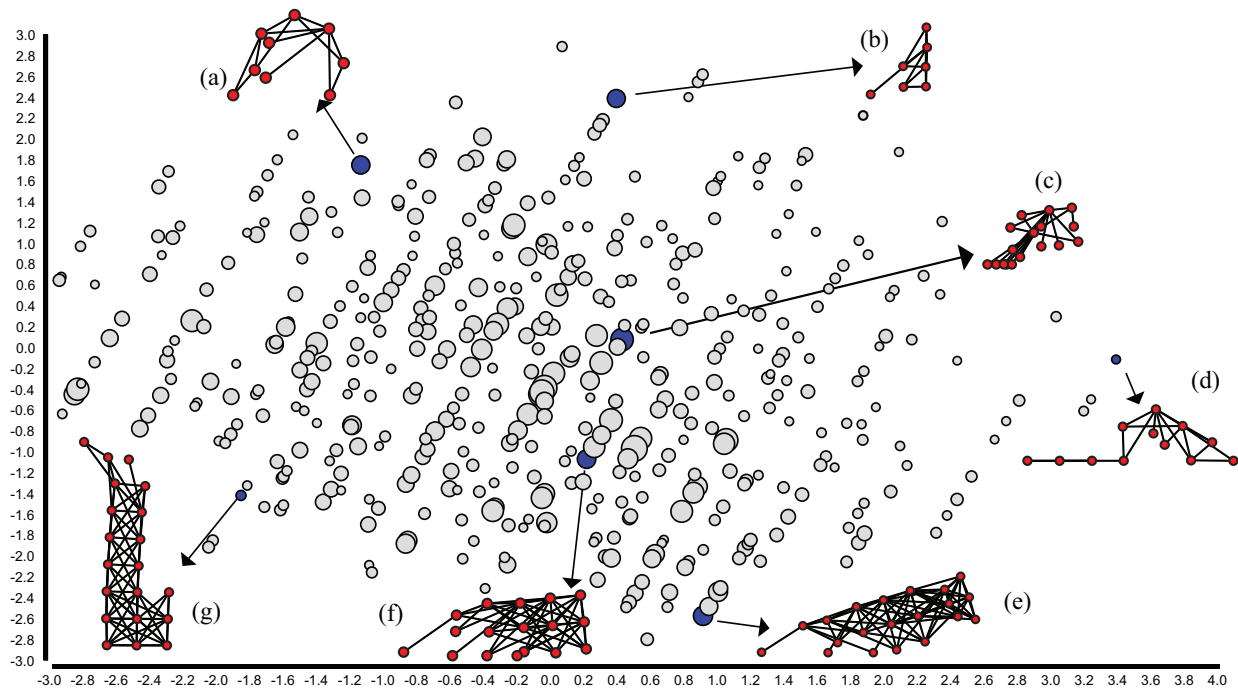


Figure 3: **PCA-based Visualization of Morphological Diversity and Performance.** The location of each point represents its respective creature morphology, while the size indicates the absolute fitness. All points shown are for creatures able to walk at least 200 units. A total of 28.1% of all 1,600 possible bins are filled with competent walkers, suggesting the diversity of ambulation methods. Furthermore, several creatures are shown to give a sense of qualitative diversity. The creatures for every point in this visualization can also be viewed in motion at <http://eplex.cs.ucf.edu/ecal13/demo/PCA.html>.

Standish, 2003; Yaeger, 1994) and diversity maintenance (Lehman and Stanley, 2011b; Mouret and Doncieux, 2012) can thus be quickly set up and conducted in the future with IESoR. Creatures can also potentially move beyond forward ambulation to more complex interactions such as foraging or predation. To facilitate such future applications, code for IESoR is available at <https://github.com/OptimusLime/IESoR>.

Conclusion

The paper demonstrated IESoR, a lightweight two-dimensional platform for evolving ambulating creatures inspired by Sodarace (McOwan and Burton, 2013, 2005). The aim is to provide an accessible platform to artificial life researchers that is inexpensive to simulate. That way, artificial life experiments that previously required significant up-front design can become easier to ramp up and build quickly. Results from searching through the indirectly-encoded creature space in IESoR with novelty search plus local competition suggest that the space indeed contains a breadth of feasible morphological discoveries with functional ambulatory capabilities, suggesting that IESoR is potentially a viable platform for artificial life research in the future.

Acknowledgements

This work was supported by the National Science Foundation under grant no. IIS-1002507 and also by a NSF Graduate Research Fellowship. Special thanks to Peter W. McOwan and Edward J. Burton for creating the original Sodarace, and to Stefan Westen for creating Mins.

References

- Auerbach, J. E. and Bongard, J. C. (2012). On the relationship between environmental and morphological complexity in evolved robots. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2012)*, pages 521–528, New York, NY. ACM Press.
- Bongard, J. C. and Paul, C. (2000). Investigating morphological symmetry and locomotive efficiency using virtual embodied evolution. In *Proceedings of the Sixth International Conference on Simulation of Adaptive Behavior*, pages 420–429. MIT Press.
- Channon, A. (2001a). *Evolutionary Emergence: The Struggle for Existence in Artificial Biota*. PhD thesis, University of Southampton.
- Channon, A. (2001b). Passing the alife test: Activity statistics classify evolution in geb as unbounded. In *Proceedings of the European Conference on Artificial Life (ECAL-2001)*. Springer.
- Clune, J., Stanley, K. O., Pennock, R. T., and Ofria, C. (2011). On the performance of indirect encoding across the continuum of regularity. *IEEE Transactions on Evolutionary Computation*.

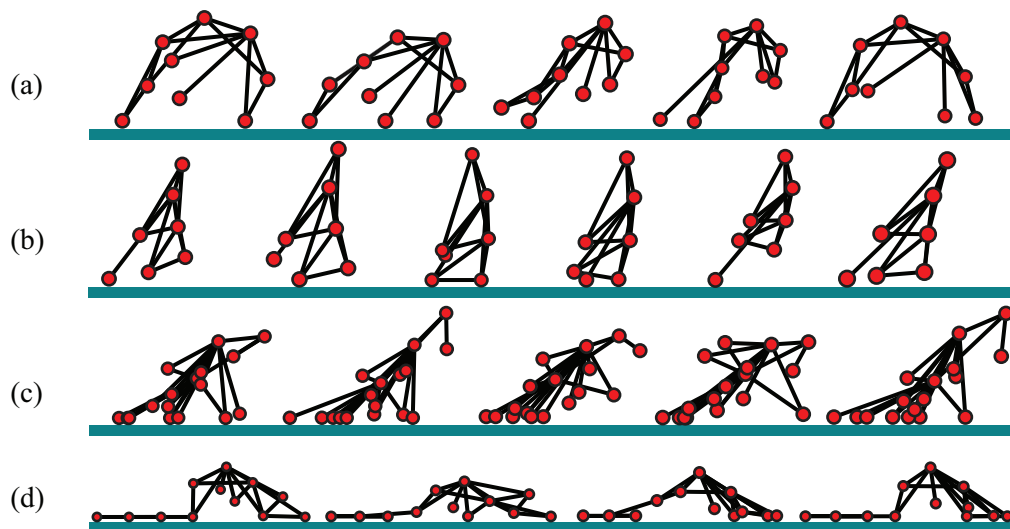


Figure 4: **Creature Motion Over Time.** The motion (from left to right over time) of a small sample of successful creatures evolved in IESoR is shown. The letters (a)–(d) correspond to those in figure 3.

- Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.
- Gauci, J. and Stanley, K. O. (2010). Autonomous evolution of topographic regularities in artificial neural networks. *Neural Computation*, 22(7):1860–1898.
- Hornby, G. S. and Pollack, J. B. (2002). Creating high-level components with a generative representation for body-brain evolution. *Artificial Life*, 8(3).
- Joachimczak, M. and Wrobel, B. (2012). Open ended evolution of 3D multicellular development controlled by gene regulatory networks. In *Proceedings of the Thirteenth International Conference on Artificial Life (ALIFE XIII)*, pages 67–74, Cambridge, MA. MIT Press.
- Krcak, P. (2007). Evolving virtual creatures revisited. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2007)*, New York, NY, USA. ACM Press.
- Lehman, J. and Stanley, K. O. (2008). Exploiting open-endedness to solve problems through the search for novelty. In Bullock, S., Noble, J., Watson, R., and Bedau, M., editors, *Proceedings of the Eleventh International Conference on Artificial Life (Alife XI)*, Cambridge, MA. MIT Press.
- Lehman, J. and Stanley, K. O. (2011a). Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2):189–223.
- Lehman, J. and Stanley, K. O. (2011b). Evolving a diversity of virtual creatures through novelty search and local competition. In *GECCO '11: Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 211–218, Dublin, Ireland. ACM.
- Maley, C. C. (1999). Four steps toward open-ended evolution. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-1999)*, volume 2, pages 1336–1343, Orlando, Florida, USA. IEEE Press.
- McOwan, P. and Burton, E. (2000–2013). Sodarace website. [URL http://sodarace.net](http://sodarace.net).
- McOwan, P. W. and Burton, E. J. (2005). Sodarace: Adventures in artificial life. In *Artificial Life Models in Software*, pages 97–111. Springer.
- Mouret, J.-B. and Doncieux, S. (2012). Encouraging behavioral diversity in evolutionary robotics: An empirical study. *Evolutionary computation*, 20(1):91–133.
- Ray, T. (1992). Evolution, ecology and optimization of digital organisms. Technical Report Working paper 92-08-042, Santa Fe Institute.
- Secretan, J., Beato, N., D'Ambrosio, D. B., Rodriguez, A., Campbell, A., Folsom-Kovarik, J. T., and Stanley, K. O. (2011). Picbreeder: A case study in collaborative evolutionary exploration of design space. *Evolutionary Computation*, 19(3):345–371.
- Sims, K. (1994). Evolving 3D morphology and behavior by competition. pages 28–39. MIT Press, Cambridge, MA.
- Standish, R. (2003). Open-ended artificial evolution. *International Journal of Computational Intelligence and Applications*, 3(167).
- Stanley, K. O. (2007). Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines Special Issue on Developmental Systems*, 8(2):131–162.
- Stanley, K. O., D'Ambrosio, D. B., and Gauci, J. (2009). A hypercube-based indirect encoding for evolving large-scale neural networks. *Artificial Life*, 15. To appear.
- Stanley, K. O. and Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10:99–127.
- Stanley, K. O. and Miikkulainen, R. (2004). Competitive coevolution through evolutionary complexification. *Journal of Artificial Intelligence Research*, 21:63–100.
- Verbancsics, P. and Stanley, K. O. (2011). Constraining connectivity to encourage modularity in HyperNEAT. In *GECCO '11: Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 1483–1490, Dublin, Ireland. ACM.
- Westen, S. (2004–2013). Mins website. [URL http://mins.sourceforge.net/](http://mins.sourceforge.net/).
- Yaeger, L. (1994). Computational genetics, physiology, metabolism, neural systems, learning, vision and behavior or polyworld: Life in a new context. In Langton, C. G., editor, *Artificial Life III, Proceedings Volume XVII*, pages 263–298. Addison-Wesley.