Division Blocks and the Open-Ended Evolution of Development, Form, and Behavior

Lee Spector
Cognitive Science
Hampshire College
Amherst, MA, USA
lspector@hampshire.edu

Jon Klein
Hampshire College, USA, *and*
Chalmers/Goteborg, Sweden
jk@artificial.com

Mark Feinstein
Cognitive Science
Hampshire College
Amherst, MA, USA
mfeinstein@hampshire.edu

ABSTRACT

We present a new framework for artificial life involving physically simulated, three-dimensional blocks called Division Blocks. Division Blocks can grow and shrink, divide and form joints, exert forces on joints, and exchange resources. They are controlled by recurrent neural networks that evolve, along with the blocks, by natural selection. Division Blocks are simulated in an environment in which energy is approximately conserved, and in which all energy derives ultimately from a simulated sun via photosynthesis. In this paper we describe our implementation of Division Blocks and some of the ways that it can support experiments on the open-ended evolution of development, form, and behavior. We also present preliminary data from simulations, demonstrating the reliable emergence of cooperative resource transactions.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; I.2.6 [**Artificial Intelligence**]: Learning—*Connectionism and neural nets*; J.3 [**Life and Medical Sciences**]: Biology and genetics

General Terms

Experimentation

Keywords

Division Blocks, breve, artificial life, open-ended evolution, morphology, development, recurrent networks

1. INTRODUCTION

In 1994 Karl Sims used evolutionary computation to produce virtual creatures, composed of rectangular blocks connected together by joints, that inhabited three-dimensional virtual worlds with simulated physics [18]. He also produced striking animations of these creatures performing a range of behaviors including swimming, walking, jumping, and following. Both the morphologies (body forms) and the neural networks that controlled them were evolved using a genetic algorithm that measured fitness relative to a desired behavior.

The project described in this paper can be thought of as an attempt to extend Sims's idea to a considerably more open-ended evolutionary and ecological context. Our goal in doing this is to allow for the exploration, in silico, of some of the key interactions between development, form, physics, behavior (including reproductive behavior), and ecology that underpin biological evolution. We are particularly interested in exploring the ways in which these factors interact, under natural selection, to produce adaptive complexity.

In the framework described here we provide no explicit behavioral fitness test. This is because the blocks that constitute our virtual creatures live or die on the basis of their acquisition and management of energy within a complex ecology. Neither do we manage reproduction and variation explicitly (as one usually does with genetic algorithms), since our blocks reproduce or fail to do so by means of evolved behaviors and using evolved mutation and recombination parameters. In contrast to Sims's creatures, which were created as adults and tested for fitness within controlled environments, our creatures (blocks or collections of blocks) must develop within their dynamic environments from "birth" to the point at which they can reproduce. And as in biology, it is sometimes unclear what should count as a single individual, or what should constitute "reproduction," within our system. This is because the connectivity of our blocks is determined by the behaviors of the blocks themselves, and because groups of blocks may sometimes form structures that are best described as collectives or as multicellular organisms.

Our goal is to provide an environment in which development, form, and behavior (including reproductive behavior) can all co-evolve in relatively open-ended ways within a simulated three-dimensional physical world. To do this we started with a physical model inspired by Sims but we then augmented it in several ways to support developmental processes and ecological interactions. Our enhanced blocks have an expanded behavioral repertoire and they are controlled by loosely constrained (evolved) recurrent neural networks that are capable of producing a wide range of developmental processes, morphologies, and ecological strategies.

The primitive elements of our framework, which we call Division Blocks, combine features of biological systems that are normally observed at different scales; some are normally

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...$5.00.

found at the cellular level, some are more characteristic of organ systems, some resemble features of colonies of microorganisms, while others resemble features of large animals with complex bodies. This mixing of scales, like the mixing of scales in Sims's creatures, can produce results that appear strange from a biological perspective, but it is justified by considerations of computational tractability. We seek an environment rich in developmental and ecological potential that is nonetheless sufficiently tractable to permit experiments on reasonably large populations over meaningful stretches of evolutionary time. We believe that our hybrid-scale framework achieves this combination and we believe that this combination, in the context of a physically simulated three-dimensional world, will permit explorations of open-ended evolution that cannot be conducted with previously developed technologies.

Several other projects have explored broadly open-ended evolution within artificial life systems but not, as far as we are aware, within a three-dimensional physical simulation environment of the sort described here. In systems such as Tierra [14], Avida [1], Echo [7] and Pushpop [20] individuals evolve in open-ended ways, managing, to some extent, their own ecological strategies and reproductive behaviors. But all of these systems "inhabit" abstract computational spaces rather than simulated three-dimensional physical environments. If the three-dimensional nature of space or the laws of physics put important constraints on biological evolution — and there is ample evidence that they do [13] — then we will learn more about evolution from artificial life systems that situate their virtual life forms within three-dimensional worlds that obey the laws of physics (in simulation or in the physical world).

Several other projects have also explored evolution within three-dimensional physical simulations, although generally with specific evolutionary goals and using highly engineered genetic algorithms or genetic programming systems. These projects have produced a variety of remarkable artifacts including robots with novel designs and means of locomotion [10, 11], structures held together by tensegrity forces [12], and even Lego block constructions [5]. Several researchers have also replicated Sims's work more directly [23], with some building systems, like Framsticks [9], intended to extend and generalize Sims's approach in a variety of ways. Bongard and Pfeifer, in their Artificial Ontogeny system [3], extended Sims's approach by adding a division-by-growth scheme similar to that described here, and by incorporating a developmental phase into their genetic algorithm.[1] But in their system individuals were evaluated in isolation, assessed relative to a pre-determined goal, and reproduced and varied according to fixed, global procedures and parameters.

One project that did couple open-ended behavioral and ecological evolution with three-dimensional simulation was the SwarmEvolve 2 project of Spector et. al [21]. But while the evolving agents in this system inhabited a three-dimensional world, they also had predetermined simple body forms (cones) so there was no possibility for the evolution of physical morphology or development. Furthermore, this system involved little physics (no gravity or elastic collisions), so few of the constraints provided by the physical world were available to guide evolution.

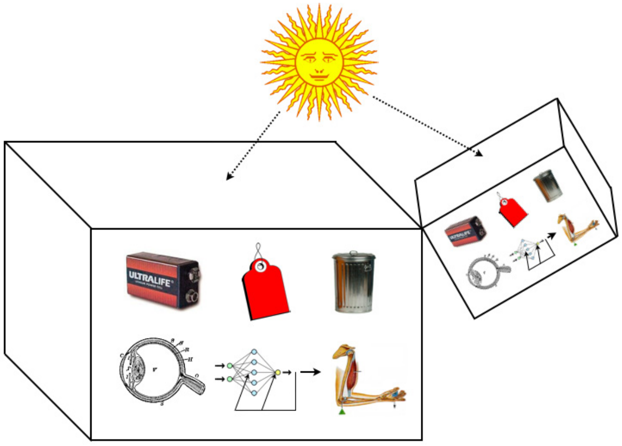

Figure 1: A schematic view of two Division Blocks, illustrating some of their features and components: variable size, joints formed during block division, photosynthetic skin, identifying tags, repositories of energy and waste, and recurrent neural networks that couple sensors to effectors.

Other systems have combined computational evolution and development in various ways, for example by evolving plant growth patterns to maximize sunlight exposure within a simulated plant ecology under a simulated sun [8], or by combining evolutionary algorithms and developmental processes to evolve "buildable" three-dimensional objects [15]. But as far as we know none of the prior work has integrated development into the open-ended evolution of form and behavior (including reproductive behavior) in complete virtual ecologies.

Why is it important that we do so? One reason is that changes in individual development are increasingly coming to be seen as critical components of evolutionary processes in general [6, 17]. Furthermore, evolutionary transitions to greater complexity seem to be linked to changes in modes of aggregation and reproduction [4, 19]. The lesson for artificial life may be that our artificial life forms will exhibit life-like evolutionary processes only when they also exhibit life-like developmental processes.

Another reason is that an artificial life system must incorporate endogenous reproduction and development, and reproduction and development must to some extent be under genetic control, if that system is to serve as a model of open-ended evolution in a complete ecosystem. Systems like Tierra and Avida provide such models but only in abstract one- and two-dimensional spaces. Division Blocks can provide models with similar properties in three-dimensional physical worlds.

In the following section we describe the principal elements of the Division Blocks framework and of our current implementation. We then present data from preliminary runs, demonstrating the reliable emergence of cooperative resource transactions. Finally, we discuss potential enhancements and additional questions about interactions between development, form, and behavior that might be explored with the aid of the system.

[1] Their scheme used spherical units that divided into pairs of smaller spheres. Spheres can be simulated more efficiently than blocks but their "division" into pairs of spheres does not conserve volume and it would create physical discontinuities in full ecological simulations.

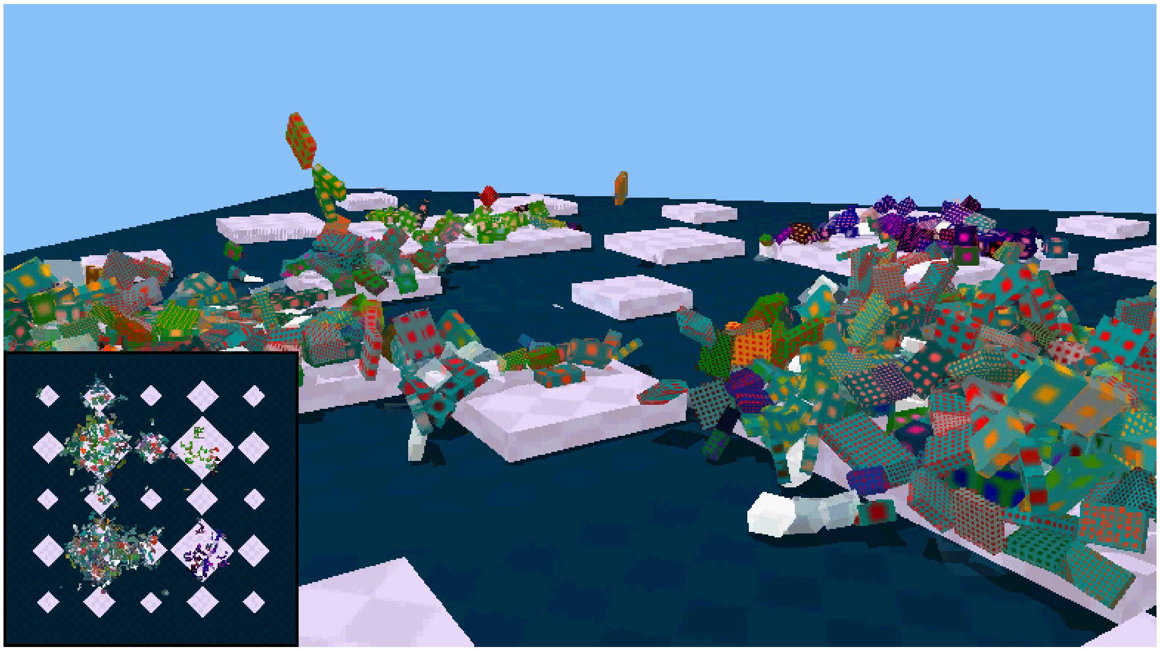

Figure 2: A snapshot of Division Blocks in a world consisting of square islands in a square ocean. A mid-range view of a portion of the world is shown in the main image and a birds-eye view of the entire world (islands and ocean) is shown in the lower left corner. In the foreground one can see blocks dying from contact with the ocean, producing expanding white block-shaped clouds of waste residue.

2. DIVISION BLOCKS

The primitive components of our virtual life forms are rectangular blocks that can grow and shrink, divide, exert forces on joints resulting from division, and conduct energy and waste transactions with other nearby blocks. They have photosynthetic skin and they may obtain energy not only from neighbors but also directly from the sun, which circumnavigates the world producing cycles of day and night. Energy is consumed by block "metabolism" (a tax on the maintenance of each cubic unit of body volume for each unit of time) and by other actions (e.g. the application of joint forces) and events (e.g. collisions). Our system of energy flow approximates biological metabolic processes only very roughly, but within this system energy is conserved and the sun ultimately provides all of the energy on which life depends. A system of "waste" flow is also modeled: metabolism produces waste as a byproduct, and accumulated waste increases the energetic cost of metabolism. Waste can be transferred among neighbors (as can energy), and it is dissipated in block-shaped clouds when blocks die (briefly blocking sunlight in the local area).

Figure 1 shows a schematic view of two connected Division Blocks, illustrating some of their most important functional components (which are described in greater detail in the following paragraphs). Figures 2 and 3 show snapshots of our system during a run. Color versions of these images, along with movies and source code for our software, can be found at the website associated with this paper.[2]

When a block divides, the two blocks into which it splits remain connected by a joint on the new faces created by the split. Forces can be applied to the joint, moving it and potentially breaking it — the threshold for breaking is under the control of the blocks. In addition, energy and waste can be passed through the joint (again, under the control of the connected blocks). In our current implementation the blocks resulting from a division event remain connected by a ball joint that is located on the corner of the division plane closest to the sun at the time of division, although many alternatives are possible both for the types of joint (e.g. they might be hinges or prismatic joints) and for their placement (e.g. they might be in the centers of the faces or in locations specified by the dividing blocks themselves). We have also considered joints that are added dynamically upon block contact (rather than by division; this would also be under the control of the involved blocks), but the system described here uses only joints produced by division.

There are also potential alternatives for the conditions under which blocks will divide. For example, block division could be entirely under the control of the blocks, permitting division at any time and in any dimension. It would probably be necessary to disallow splits that would produce edge lengths below some pre-specified minimum, since otherwise the system might produce huge numbers of extremely small blocks, presenting difficulties both for physical simulation and for visualization. This and other schemes deserve exploration, but in our current implementation we have opted for a simpler size-based division scheme that was inspired by the behavior of some kinds of biological cells: when any dimension of a block exceeds a certain pre-specified maximum then the block divides along that dimension (producing a joint on the corner of the division plane closest to the sun).

When a single, un-jointed block divides, the new joint is designated as the "stem joint" of both of the new child blocks that are produced. When a block that already has a

[2] http://hampshire.edu/lspector/db-gecco-2007

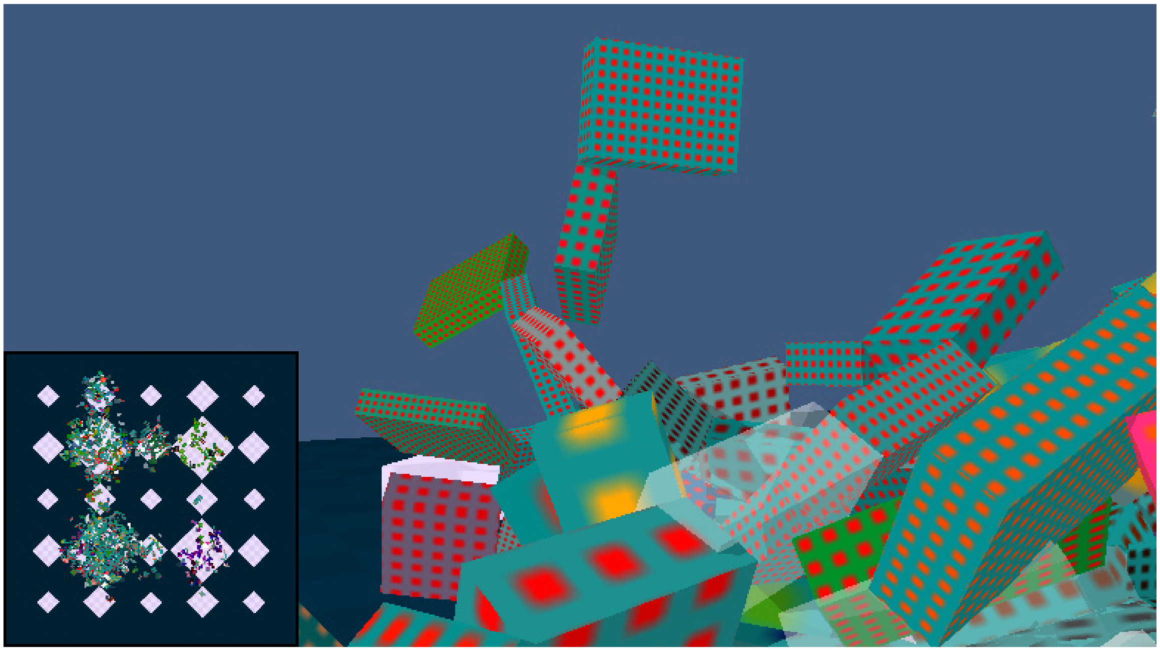

Figure 3: A snapshot of Division Blocks taken slightly later in the same run that produced Figure 2. This snapshot shows a close-up view of a portion of the world in the main image and a birds-eye view of the entire world (islands and ocean) in the lower left corner. The ghostly, translucent block in the center is the dissipating residue of waste from a block that recently died; we can see through it, but it is opaque to the simulated light that drives photosynthesis.

stem joint divides the designated stem joint of one child (the one still connected to the existing stem) will be the existing stem, while the designated stem joint of the other child will be the new joint that is formed by the division. This system of stem designation permits the expression of asymmetric sensing and action strategies, since block sensors and effectors (described in detail below) generally come in pairs, one for stem connections and one for non-stem connections.

Most ecological transactions between blocks are mediated by a block-recognition scheme based on "tags" [7, 16]. Each block has a dynamically modifiable tag, which is just a floating point number, and dynamically changeable tolerances (also floating point numbers) for various kinds of transactions. In general the transaction will be permitted by a particular block if the difference between the tags of the two participants is less than the relevant tolerance of the block in question. Tags and tolerances, all under the control of the blocks, mediate energy transfers, waste transfers, and sexual recombination (although in the case of recombination each block specifies a "mate tag" that is independent of its main tag, permitting strategies based on out-breeding). Transactions are generally consensual in the sense that it is up to each block (expressed via tag and tolerance values) whether or not it will donate energy or accept waste in any interaction. The sizes of energy donations and waste collections are also under the control of the participating blocks — the amount transferred is the average of the sizes specified by the two participating blocks. Transactions may happen (stochastically) at each joint in each time step of the simulation, and they may also happen upon each collision.

Each block also contains a sine-wave "pulse" oscillator, the frequency of which is controlled by the block, that can be incorporated into the control strategy of the block in a wide variety of ways. This is similar to the "oscillate-wave" nodes in Sims's system, which also provided a saw-tooth oscillator and a variety of other high-level signal processing components that might be incorporated into future Division Blocks systems [18].

The control system of each block is a recurrent neural network with nodes for each sensor input, each effector output, and some fixed number of "hidden" nodes. We used ten hidden nodes in the experiments presented below. The activation values of nodes range from -1 to 1. The structure of the neural network is expressed as a matrix of connection strengths that are also real numbers between -1 and 1.[3] At each time step each node produces its next activation value (and output) by summing its inputs (the activations of all other nodes weighted by their connection strengths), passing the sum through a sigmoid activation function, and then calculating a weighted average of the resulting value and the node's previous activation. The weighted averaging smoothes the transitions of activation values in the network, providing a sort of activation "persistence"; we currently weight the new value at 25% and the prior value at 75%.[4] The steepness of the sigmoid activation function is itself under the control of the block, via the formula:

[3] We provide an option to prohibit direct connections from inputs to outputs, requiring that such connections be mediated by hidden nodes. This is done by ensuring that the strengths for direct connections always have a value of zero. In the experiments presented here, however, we did not use this option, so any recurrent architecture was permissible.

[4] A similar mechanism, called "inertia," is used in the neural networks that control Framsticks [9].

Table 1: Sensors available to each Division Block. All values are scaled to $[-1, 1]$ if necessary.

Sensor	Description
zero, plus, minus	Always provides a value of zero, $+1$, or -1.
energy	The energy reserves of this block.
waste	The amount of waste accumulated in this block.
exposure	The amount of sunlight incident on this block.
pulse	The current state of this block's sine wave oscillator.
rotx, roty, rotz	The x, y, and z components of this block's rotation relative to the world coordinate frame.
localtag	The average tag value of blocks in this block's neighborhood.
localenergy	The average energy level of blocks in this block's neighborhood.
localwaste	The average amount of waste in blocks in this block's neighborhood.
connectedtag	The average tag of blocks joined to this block by non-stem joints.
connectedenergy	The average energy of blocks joined to this block by non-stem joints.
connectedwaste	The average waste of blocks joined to this block by non-stem joints.
stemtag	The tag of the block joined to this block via the stem joint (if any; otherwise 0).
stemenergy	The energy of the block joined to this block via the stem joint (if any; otherwise -1).
stemwaste	The waste of the block joined to this block via the stem joint (if any; otherwise -1).

$$\sigma(s) = \frac{2}{1 + e^{-cs}} - 1.$$

The variable s here is the weighted sum of the node's inputs, while c is a compression (steepness) factor obtained from `sigmoidcompression` (one of the network's effector outputs, which can range from -1 to 1) and a pre-specified minimum and maximum (currently $min = 0.1$ and $max = 100$) by the formula:

$$c = min + \left(\frac{sigmoidcompression + 1}{2} \times (max - min)\right)$$

In general the new states of a block's effector nodes will influence its behavior immediately, although in the case of the effectors controlling physical size we found it necessary to provide additional limits on rates of growth (currently limited to a scaling of 101% per time step) and shrinkage (currently limited to a scaling of 99% per time step) to prevent radical physical dislocations. This seems reasonable because the mere "willing" of a size change by a block's neural network should not be expected to produce the full size change in a single moment; objects in the simulated world, as in the physical world, can only respond gradually. Our system also allows one to charge metabolic costs for size changes, although in the experiments presented below we do not do so.[5]

When a block divides, each new child receives a possibly-modified version of the neural network from the now-divided parent. The possible sources of modification are mutation and sexual recombination. In mutation, which occurs for each child block for each division, each element of the neural network's connection strength matrix may be perturbed, with a probability derived from a `copyfidelity` effector output of the parent. The size of the perturbation, if it occurs, is limited by a `mutationlimit` effector output of the parent. In other words, the parent can control both the probability that each connection strength will be mutated at all and the maximum size of such perturbations. The actual copy fidelity values and mutation limits used during inheritance are determined from the network outputs using formulae similar to that for sigmoid compression (above), with boundary conditions that permit highly accurate but not perfect reproduction. In the experiments presented below, copy fidelity could range from zero (meaning that every connection strength would be perturbed) to $50,000$ (meaning that only one in $50,000$ connection strengths would be perturbed), and the mutation limit could range from 0.1 to 1.0. Interestingly, in our experiments to date we have observed the evolution of replication that is reasonably accurate but we almost never observe the emergence of the maximal replication accuracy permitted by our boundary conditions; intermediate levels of replication accuracy are apparently adaptive.

New blocks formed by division are "born" with their effector nodes at default levels of activation (zero, which is the middle of their range), but the activation values of the hidden nodes of the child networks are inherited from the parent network in an asymmetric way: one child (chosen in an arbitrary but consistent way) inherits the activations of the parent's odd-numbered hidden nodes, while the other child inherits the activations of the parent's even-numbered hidden nodes. Nodes that do not inherit values are set to zero. This mechanism of asymmetric state inheritance permits, at least in theory, strategies of differential development that are independent of the sensory environments of the child blocks. While we have not yet measured the effects of this mechanism we hope that it can support, in a primitive way, essential elements of development that are akin to the processes of biological cell differentiation during embryogenesis.

The sensors available to each block, which are detailed in Table 1, provide information not only about the block itself but also about those to which it is connected and others within a small neighborhood (for which we used a radius of about four average block widths). The effectors available to each block are detailed in Table 2. Some of the effector values require scaling before application in the simulated environment, as with the `copyfidelity` and `mutationlimit` effectors described above; full details are available in the source code at the website associated with this paper.

For the purpose of visualization we color the block skins with patterns that can reveal up to seven internal state vari-

[5] Of course there is still a cost for size itself — the metabolic charge for each unit of body volume — but we do not charge separately for size changes in the experiments reported here.

Table 2: Effectors available to each Division Block. All values are scaled from $[-1, 1]$ to appropriate ranges if necessary (see text).

Effector	Description
sizex, sizey, sizez	Target x, y, and z sizes for this block.
jointx, jointy, jointz	x, y, and z components of the target joint position for non-stem joints.
stemx, stemy, stemz	x, y, and z components of the target joint position for the stem joint (if any).
tag	The tag of this block.
donationsize	Size of energy donations that this block will make to non-stem others.
donationtolerance	Maximum tag difference through which this block will donate energy to non-stem others.
stemdonationsize	Size of energy donations that this block will make to its stem.
stemdonationtolerance	Maximum tag difference through which this block will donate energy to its stem.
collectionsize	Size of waste collections that this block will take from non-stem others.
collectiontolerance	Maximum tag difference through which this block will collect waste from non-stem others.
stemcollectionsize	Size of waste collections that this block will take from its stem.
stemcollectiontolerance	Maximum tag difference through which this block will collect waste from its stem.
copyfidelity	Probability that a neural network connection strength will be inherited without mutation.
mutationlimit	Maximum size of mutations to neural network connection strengths during division.
matecontribution	Probability that a connection strength will be taken from a neighbor rather than the parent.
matetag	Preferred tag for recombination partner; neighbor with tag closest to this will be chosen.
adhesion	Joint strength is the average adhesion value from the two joined blocks. If the forces on the joint exceed this then the joint will break.
pulserate	Frequency of this block's sine wave oscillator.
sigmoidcompression	Compression parameter to the sigmoid activation function used for this block.

ables. We use a dotted texture pattern and encode one variable in the density of the dots and six variables in the red, green, and blue color channels of the dot and of the background field. In the experiments described here we displayed energy via dot density, with a dense pattern of small dots indicating high energy, a sparse pattern of large dots indicating low energy, and intermediate densities indicating intermediate levels of energy. We used the red channel of the dots to display light exposure, the green channel to display the (non-stem) waste collection tolerance, and the blue channel to display the (non-stem) waste collection size. We used the red channel of the background field to display level of waste, the green channel to display (non-stem) energy donation tolerance, and the blue channel to display (non-stem) energy donation size.

The floor on which our blocks reside can be configured in a variety of ways. For the experiments described here we used a pattern of small square "island" floors of various sizes, situated within a square "ocean" that cannot support blocks; blocks touching the ocean are charged a large energy cost, and blocks landing on the ocean will sink into it and be consumed. Islands limit the sizes of local populations and seem to support richer patterns of diversity. Collections of blocks can spread from island to island either by reaching long "arms" of joined blocks from one to another or by "flinging" blocks into the air (via strong joint forces that break their joints) in ways that send them long distances.

At the beginning of a run we create new blocks in random locations and with random neural networks, and the large majority of these blocks fail to effectively manage their resources, to grow, and to divide into child blocks that are themselves viable. We therefore continue to pump random blocks into the world until a self-sustaining, "reproductively competent" population is produced. In the experiments described below we set the threshold for reproductive competence to 250; whenever there were fewer than this number of blocks new random blocks would be added to the world. Note that energy is not conserved prior to reproductive competence; we collected data and drew inferences from the behavior of the system only well after reproductive competence had been established.

Our implementation includes additional features that cannot be documented fully here, many of which are intended to enhance ecological realism in ways that may influence the emergence of adaptive complexity. For example we model a local resource that is roughly analogous to air, associating an energy cost with crowding. We also include mechanisms to produce mountain-like barriers, to model catastrophic events such as comet impacts and epidemics, and to link mutation rates to sun exposure. We have described the system's most characteristic features here; for full details please refer to the materials at the website associated with this paper. Our implementation of Division Blocks runs within the open source breve simulation environment.[6]

3. PRELIMINARY RESULTS

Experiments with Division Blocks produce large volumes of data that can be used, in conjunction with the systematic variation of parameters, to explore a variety of evolutionary hypotheses. In this paper we present quantitative results primarily as an illustration of the type of data that can be obtained, and we do so only in the context of a single set of system parameters (although we report on results from many independent runs). Nonetheless, the data from this set of runs exhibit some interesting features that may have more general implications.

We conducted 40 independent runs of our implementation using the parameter settings described above.[7] We configured the system to collect and print population-wide block state data every 10 time steps, and we calculated statistics

[6] `http://www.spiderland.org/breve`

[7] Values of parameters not discussed explicitly above were as specified in the source code available at the website associated with this paper.

on the basis of the first data that was printed after 1000 time steps of reproductive competence (see above). The average age of blocks at the time of this reporting, averaged over all 40 runs, was 39.1 time steps, so our data generally reflect the state of the system from at least 25 generations beyond the achievement of reproductive competence. The average time step at time of reporting was 6911, meaning that it took nearly 6000 time steps, on average, to achieve reproductive competence. The average number of blocks at the time of reporting was 392.

Figure 4 shows some of this data, displaying the cross-run averages of the population-wide averages of seven variables of interest. Error bars indicate the range of variation (one standard deviation above and below) across the 40 runs. The plot marked "A" displays the average tag values which, as expected, are widely distributed with an average value, across all runs, of about 0.0 (the center of the range). There is no reason for any particular tag value to be more generally adaptive than any other, so what we see here is a random distribution across runs.[8] Plot A therefore provides a standard to which other plots can be compared; any which differ significantly from A probably reflect adaptations.

The plot marked "B" displays the average values of effectors that control energy donation: `donationsize` (on the left) and `donationtolerance` (on the right). The relatively high values for these variables indicate that most blocks, in most runs, are donating energy in relatively large quantities and without much discrimination. The relatively narrow range of variation of these variables indicates that this cooperative energy donation behavior emerges fairly reliably. Why might this be the case? Possibly because donations produce disparities that increase the chances that one member of a pair of connected sibling blocks will survive to divide again, or for other reasons that have been discussed in the literature of the evolution of altruism (e.g. [16]); more research will be required to sort this out.

The plot marked "C" displays `stemdonationsize` (on the left) and `stemdonationtolerance` (on the right). The fact that these values are lower than those for non-stem donation indicate a tendency for blocks to adopt asymmetric donation strategies, although the stinginess toward stem blocks is less reliable than the altruism toward non-stems. Plot D displays the average values of `matecontribution`, demonstrating a strong and reliable tendency against sexual recombination, at least at this stage of a run and with the parameters that were used. Plot E displays the average value of `adhesion`, indicating the high adaptive value of strong joints in the system as it was configured.

One can speculate about adaptive explanations for each of these results, but as with the energy donation results in plot B a variety of explanations may be available and additional experiments would be required to draw firm conclusions from plots C–E . The important point at this stage, however, is that the Division Blocks system produces data such as these that can be compared among differently parameterized runs; for example one might explore theories about the adaptive benefits of recombination by changing various parameters and by observing changes to plot D.

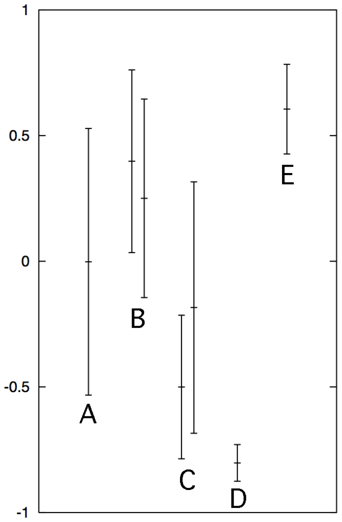

Figure 4: Averaged data from 40 runs of the Division Blocks system, collected after 1000 time steps of reproductive competence. Error bars indicate ±1 standard deviation. A: average tag values; B: average `donationsize` (left) and `donationtolerance` (right); C: average `stemdonationsize` (left) and `stemdonationtolerance` (right); D: average `matecontribution`; E: average `adhesion`.

4. DISCUSSION

There is much more that can be explored with Division Blocks than we have been able to present here, even in the context of the current implementation and even with a single set of configuration parameters. For example it would be interesting to track the variables shown in Figure 4, along with other variables (including the numbers, sizes, and distributions of blocks, the other effector values in Table 2, etc.), across long periods of evolutionary time. It would also be interesting to analyze the behavior of the system relative to measures of "evolutionary activity" that have been described in the literature [2, 22]. Additional insights might be gleaned from variation of environmental conditions and from incorporation of extensions such as the ability to form new joints when blocks collide. An extension that is currently under development involves parallelizing the system by allowing block transport between simulations that are running asynchronously on many computers in a high-performance cluster. Simulations with over one or two thousand blocks become unmanageably slow on current desktop hardware, but with cross-simulation block transport (occurring, for example, at the edges of the simulated world) we should be able to connect many machines to support simulations with numbers of blocks that are bounded only by the amount of available hardware.

The really intriguing possibilities for future work, however, concern ways in which Division Blocks might be used to explore new questions about the relations between development, form, and behavior in open-ended evolutionary processes. For example, what environmental conditions are required for the emergence of various kinds of morphological or developmental complexity? What patterns of resource

[8] Tag distributions *within* runs may not be random and they may display complex dynamics involving drift, founder effects, adaptation, etc. But because no tag value has any intrinsic meaning we would expect the tag values involved in such dynamics to be distributed randomly across runs.

sharing and flow — involving resources such as sunlight, space, energy, and waste — accompany the emergence of such complexity? Under what physical and ecological conditions do collective activity and multicellularity emerge?

One more radical direction for future work concerns the possibility of building truly physical (not simulated) Division Blocks. This may be possible, in principle, because the Division Blocks framework is grounded on the principle that energy is conserved. Of course, energy conservation is only crudely approximated in the current simulation, and many features of the current Division Blocks framework would present extreme engineering challenges for physical construction. Systems of self-organizing physical blocks have recently been demonstrated, however [24], and it is possible that such systems could be enhanced to incorporate the key features of the Division Blocks framework: growth, division, and inheritance of behavior. Such developments are a long way off at present, but their possibility may nonetheless serve to guide continuing experiments in simulation.

5. ACKNOWLEDGMENTS

Jason Tor, Ray Coppinger, Chris Perry, Kyle Harrington, Andrew Stout, and anonymous reviewers provided suggestions that helped to improve this work. Members of the Hampshire College School of Cognitive Science provided feedback on a preliminary presentation, and Rebecca Neimark edited and improved the draft. This material is based upon work supported by the U.S. National Science Foundation under Grant No. 0308540. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] C. Adami and T. C. Brown. Evolutionary learning in the 2d artificial life system avida. In *Artificial Life IV*, pages 377–381, Cambridge, MA, 1994. MIT Press.

[2] M. A. Bedau, E. Snyder, and N. H. Packard. A classification of long-term evolutionary dynamics. In *Artificial Life VI: Proc. of the Sixth Int. Conf. on Artificial Life*, pages 228–237, Cambridge, MA, 1998. The MIT Press.

[3] J. Bongard and R. Pfeifer. Evolving complete agents using artificial ontogeny. In F. Hara and R. Pfeifer, editors, *Morpho-functional Machines: The New Species (Designing Embodied Intelligence)*, pages 237–258. Springer-Verlag, Berlin, 2003.

[4] J. T. Bonner. *The evolution of complexity by means of natural selection*. Princeton University Press, Princeton, NJ, USA, 1988.

[5] P. Funes and J. Pollack. Evolutionary body building: Adaptive physical designs for robots. *Artificial Life*, 4(4):337–357, Fall 1998.

[6] G. Gottlieb. *Individual development and evolution: The genesis of novel behavior*. Oxford University Press, New York, 1992.

[7] J. H. Holland. *Hidden order: how adaptation builds complexity*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1995.

[8] C. Jacob. *Illustrating Evolutionary Computation with Mathematica*. Morgan Kaufmann, 2001.

[9] M. Komosinski and A. Rotaru-Varga. Comparison of different genotype encodings for simulated 3D agents. *Artificial Life Journal*, 7(4):395–418, Fall 2001.

[10] H. Lipson and J. B. Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978, August 2000.

[11] I. Macinnes and E. Di Paolo. Crawling out of the simulation: Evolving real robot morphologies using cheap reusable modules. In *Artificial Life IX: Proc. Ninth Intl. Conf. on the Simulation and Synthesis of Life*, pages 94–99, Cambridge, MA, 2004. MIT Press.

[12] C. Paul, H. Lipson, and F. J. V. Cuevas. Evolutionary form-finding of tensegrity structures. In *Proc. 2005 Conference on Genetic and Evolutionary Computation*, pages 3–10, New York, NY, 2005. ACM Press.

[13] R. Pfeifer and J. Bongard. *How the Body Shapes the Way We Think: A New View of Intelligence*. MIT Press, Cambridge, MA, USA, 2006.

[14] T. S. Ray. An approach to the synthesis of life. In *Artificial Life II*, pages 371–408, Redwood City, CA, 1992. Addison-Wesley.

[15] J. Rieffel and J. Pollack. Automated assembly as situated development: using artificial ontogenies to evolve buildable 3-d objects. In *Proc. 2005 Conference on Genetic and Evolutionary Computation*, pages 99–106, New York, NY, 2005. ACM Press.

[16] R. L. Riolo, M. D. Cohen, and R. Axelrod. Evolution of cooperation without reciprocity. *Nature*, 414(6862):441–443, November 2001.

[17] C. D. Schlichting and M. Pigliucci. *Phenotypic Evoution: A Reaction Norm Perspective*. Sinauer Associates, Inc., Publishers, Sunderland, MA, USA, 1998.

[18] K. Sims. Evolving virtual creatures. In *SIGGRAPH '94: Proc. 21st Annual Conference on Computer Graphics and Interactive Techniques*, pages 15–22, New York, NY, USA, 1994. ACM Press.

[19] J. M. Smith and E. Szathmáry. *The origins of life: From the Birth of Life to the Origin of Language*. Oxford University Press, Oxford, UK, 1999.

[20] L. Spector. Adaptive populations of endogenously diversifying Pushpop organisms are reliably diverse. In *Proc. Artificial Life VIII, the 8th Intl. Conf. on the Simulation and Synthesis of Living Systems*, pages 142–145. The MIT Press, 2002.

[21] L. Spector, J. Klein, C. Perry, and M. Feinstein. Emergence of collective behavior in evolving populations of flying agents. *Genetic Programming and Evolvable Machines*, 6(1):111–125, Mar. 2005.

[22] A. Stout and L. Spector. Validation of evolutionary activity metrics for long-term evolutionary dynamics. In *Proc. 2005 Conference on Genetic and Evolutionary Computation*, pages 137–142, New York, NY, USA, 2005. ACM Press.

[23] T. Taylor and C. Massey. Recent developments in the evolution of morphologies and controllers for physically simulated creatures. *Artif. Life*, 7(1):77–87, 2000.

[24] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson. Self-reproducing machines. *Nature*, 435(7039):163, 2000.