

# Towards an evolutionary-developmental approach for real-world substrates

Shivakumar Viswanathan and Jordan Pollack  
DEMO Lab, Computer Science Dept., Brandeis University, MA 02454  
{shiva,pollack}@cs.brandeis.edu

## Abstract

Extending “body-brain” evolution to the real-world presents a number of difficulties due to conflicting idealizations between evolutionary and constructional models. Toward addressing this gap, we develop a simple model system to analyze the effects of undoing these idealizations. Preliminary experiments with this system show that high variability developmental substrates can influence evolutionary dynamics by causing ambiguities in selection. Furthermore the substrate can enable the evolution of adaptive responses to non-deterministic developmental effects.

## Introduction

### Background

An important approach towards Artificial Life is via the investigation of the system principles of interaction and self-organization by building robots (Brooks, 1992). A key principle being to design the robots such that their adaptive behavior is achieved with a parsimony of global knowledge models (Brooks, 1991; Harvey et al., 1997; Pfeifer, 1996). The rationale being that the structure of the systems required to achieve the desired behavior under this constraint could provide insights into the nature of interaction-based adaptive behavior both in biological as well as engineering systems. This rationale in turn suggests that the same principle of global knowledge parsimony could be applied to the design of the very processes by which the entire robots (i.e. “body and brain”) themselves come to be (Sims, 1994; Pollack et al., 1999).

We can distinguish two interpretations of the general question of how complete adapted robots could be obtained by interaction based processes. From a *functional* perspective, the explanandum is of how adapted robots could be evolved from unadapted ones with respect to a particular behavioral context. From an *embodiment* perspective, the explanandum is how particular physically embodied robots could be constructed (or transformed) from simpler precursors by such processes. When the solution concept of interest is in terms of interaction-based processes that can gener-

ate actual embodied robots performing the desired behavior<sup>1</sup> both these aspects require to be addressed.

### Motivation

In industrial engineering practice, knowledge intensive protocols of abstract specification and validation enable the functional and embodiment aspects to be addressed with a significant degree of independence, as problems of design and manufacturing respectively. Applying the principle of knowledge parsimony, the question then is how could these aspects be addressed in an integrated way without making the knowledge intensive specification/implementation distinction. Even though such a distinction is not involved in biological evolution, we believe that achieving this integration in an artificial evolutionary system presents some basic conceptual difficulties.

In evolutionary algorithms with an explicit developmental phase<sup>2</sup>, the embodiment aspects are treated as a deterministically unfolding process that satisfies the Genotype-Phenotype map abstraction. Though this abstraction is a key axiom in Evolutionary Computation, development as a generative representation of the phenotype is far removed from the issues involved in the real-world construction of complex structures.

A more relevant, even if simplistic, characterization of problems related to embodiment have been discussed by (Simon, 1962) and (Crane, 1950). In the parable of the two watchmakers, Simon (Simon, 1962) discusses how problems caused by the instability of intermediate stages in constructing complicated structures depends on the structure of the construction process even though the specifications of the final structure are known to both watchmakers. Similarly, Crane (Crane, 1950) discusses the problem of error accumulation in the assembly of structures involving a large number of parts and how the specificity of interactions dur-

---

<sup>1</sup>As differ from a solution concept in terms of build-able specifications for such robots as in (Funes and Pollack, 1998; Lipson and Pollack, 2000; Hornby and Pollack, 2002)

<sup>2</sup>See (Hornby and Pollack, 2002; Stanley and Miikkulainen, 2003) for extensive reviews

ing construction has an impact on this accumulation. However, in both these examples the functional aspects are neglected. There is an implicit idealization of a goal structure where processes unable to produce this specific outcome are categorically maladapted. Furthermore there is no notion of how these processes may be systematically varied to retain these properties while producing different final structures.

This suggests the existence of a gap in the way the way that the functional and embodiment aspects of biological systems has been conceived, where each is typically studied by idealizing the other. The conundrum is that with respect to the problem of producing complete robots neither of these idealizations is entirely valid. In this paper we discuss preliminary investigations toward identifying concepts required to bridge this functional-embodiment divide within an evolutionary developmental framework.

## Problem definition

### Basic model

In order to model both functional as well embodiment aspects, we adopt a simple evolutionary model and introduce modifications to the developmental phase to accommodate issues relevant to embodiment processes. This is described in an implementation independent manner below.

Development is considered to be a series of transformations starting with the zygote and terminating at the adult organism which enters the reproducing population on which selection acts. The process occurs in a particular environmental context  $\zeta$  and  $\Phi_\zeta$  is the set of all phenotypes. The zygote is defined as  $\phi_{start} \in \Phi_\zeta$ , and the net effect of the development process is to produce a series of intermediate states of the embryo  $\phi_{start} \rightarrow \dots \phi_t \rightarrow \phi_{t+1} \rightarrow \dots$  till it halts at the “adult” phenotype  $\phi_{final}$ . This temporally ordered sequence of intermediate phenotypic states from  $\phi_{start}$  to  $\phi_{final}$  constitutes the ontogeny of  $\phi_{final}$ . The final configuration  $\phi_{final}$  obtained on a given execution instance is taken to be the individual which enters the reproducing population. The behavioral measure of interest (i.e. the fitness function) is defined to be  $\mathbf{e} : \Phi_\zeta \rightarrow \mathbb{R}$ .

The zygote has associated with it a genotype  $g \in \mathcal{G}$  and the cellular interpreter machinery  $\mathbf{i} \in I$  for the genotype ( $\mathcal{G}$  is the set of all genotypes and  $I$  is the set of all interpreters). The interaction of  $g$  and  $\mathbf{i}$  results in “actions” that effect the present state of the developing embryo. The various interactions involved in development are assumed to be effected by ambient variability that has a stable pattern particular to the operative physics and  $\zeta$ . So development with the same zygote  $\phi_{start}$  would not necessarily result in the same outcome.

Consequently, rather than a genotype-phenotype map we have a relation  $\mathcal{D} \subset \mathcal{G} \times \Phi_\zeta$ , where a pair  $(g, \phi) \in \mathcal{D}$  indicates the possible genotype and phenotype combination for a given individual. So each genotype  $g$  can be considered to be associated with a finite set (sample space)  $\Omega_g$  of possible adult phenotypes where  $\Omega_g \subset \Phi_\zeta$ . A phenotype  $\phi \in \Phi_\zeta$

is in  $\Omega_g$  iff  $\phi$  has a non-zero probability of being produced as  $\phi_{final}$  with  $g$ . Formally the development process can be described as a function  $\delta_{\phi_{start}} : \mathcal{G} \rightarrow \tilde{\Omega}$  where  $\tilde{\Omega}$  is the set of pairs  $(\Omega_g, P_g)$  for all  $g \in \mathcal{G}$ , with  $P_g$  being the probability distribution on  $\Omega_g$ .

### Selection for specificity

Consider the sample space  $\Omega_g$  associated with a genotype  $g$ . *Specificity* is used here with respect to the maximum possible difference in fitness between two phenotypes in  $\Omega_g$ . Suppose  $e_{min}$  and  $e_{max}$  are the minimum and maximum values of  $\mathbf{e}$  over the phenotypes in  $\Omega_g$ . A closed interval  $I_{\Omega_g} = [e_{min}, e_{max}]$  is defined on the real-number line using these two values, where  $e_{min}$  and  $e_{max}$  are the greatest lower bound (g.l.b) and least upper bound (l.u.b.) of  $I_{\Omega_g}$  respectively ( $e_{max}, e_{min} \neq \pm\infty$ ). So the specificity associated with  $g$  is the length of the interval  $s = |I_{\Omega_g}| = e_{max} - e_{min}$ .

So we can say that the developmental process has high specificity if  $s$  is a “small” value and low specificity if  $s$  is a “large” value. In general terms, a process that could produce individuals having high fitness values with high specificity even in the presence of ambient variability is very desirable as a solution.

With respect to an evolutionary process though, it is a higher-order property as compared to the fitness function  $\mathbf{e}$  based on the behavior of the individuals in the population. So a question of interest is whether there would be an implicit selection for genotypes that have greater *specificity* of fitness outcomes especially when associated with high values of both  $e_{min}$  and  $e_{max}$ . This is also question of interest with respect to biology where developmental processes are known to have high reproducibility in their outcomes. It raises a chicken-egg question of whether developmental specificity is due to properties unique to the living or whether the specificity was an implicitly evolved property. Here we use evolution on a simple toy-system to demonstrate how developmental substrates capable of “measurement” could influence the evolution of specificity (Viswanathan and Pollack, 2004).

## Development with a tiling machine

**Implementing genetic “actions”** The toy system described in this section instantiates the model described in the previous section. The “actions” resulting from the interpretation of the genotype, take the form of assembly actions. These are performed by a tiling machine modeled as a gantry robot that is restricted to movement in two dimensions (see Figure 1).

This machine has a head that can be moved under programmable control to locations in the workspace identified by their  $(x, y)$  coordinates, by a series of horizontal and vertical translations. The workspace, which is the equivalent of the environmental context  $\zeta$ , is a square partition of a two dimensional plane such that it can perfectly accommodate

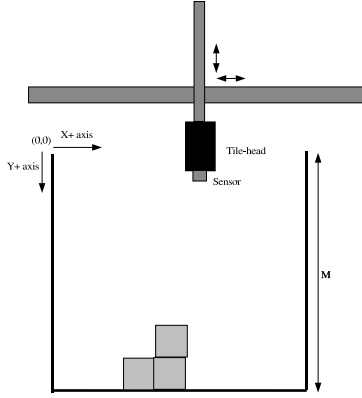


Figure 1: Tiling machine in workspace  $\zeta$

$M \times M$  identical square tiles without any gaps. The machine has an explicit spatial existence so it cannot move through tiles.

The atomic operations performed at the head are a **release** operation to release tiles individually, and a **sense** operation by means of a sensor that returns a binary outcome based on whether the position immediately in front of it in the  $Y+$  direction is occupied or not with a **1** or **0** respectively. A finite number of tiles are available to the machine which can be released individually at the head under programmable control.

This programmed control is achieved by means of a program  $g$  (the genotype) which is executed with a fixed interpreter  $\mathbf{i}$  embedded in the tiling machine, where a program is a finite sequenced list of  $(x, y)$  locations in the workspace. From this we can see that configuration of tiles in the workspace at time  $t$ , is the equivalent of the state of the embryo  $\phi_t$ . And with the execution of the program  $g$ , the tile configurations also change resulting in a developmental process<sup>3</sup>

**Tile physics** The physics governing the behavior of a tile on being released at the tile-head is similar to the Tetris game. There is a constant velocity “diffusion” acting downward along the  $Y+$  direction such that a tile released at a particular location moves at the rate of one tile length per time step. The stochasticity arises from the interference when a released tile comes in contact with other tiles or with the edges of the workspace and continues till the tile comes to rest.

<sup>3</sup>The motion from the origin to a point  $(x_a, y_a)$  takes place by horizontal motion along the  $y = 0$  edge till the  $x_a$  coordinate is reached, followed by downward motion till the location is reached if unobstructed. This path is retraced to return to the origin. A movement between consecutive locations occurs indirectly by first returning to the origin.

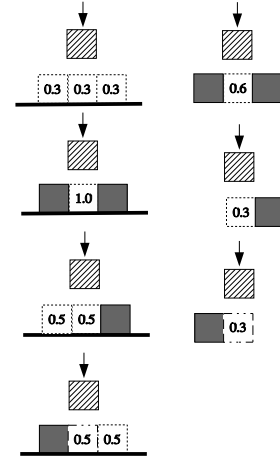


Figure 2: Tile interference physics

The interference is modeled as occurring as defined in Figure 2. The hatched square is the tile that is descending. Its possible position in the next time step are shown as squares with dotted outlines. The numbers indicate the probability with which the tile will come to rest at that position. The particular values chosen are arbitrary. The dark line indicates a tile or a wall at that periphery and the tile cannot proceed through it.

**Interpreters** The focus in this setup is on the effect of features in the genotype-interpreter relation. Here we consider two interpreters  $\mathbf{i}_{coupled}$  and  $\mathbf{i}_{state}$ . The interpreter  $\mathbf{i}_{coupled}$  is such that for each location in the program, the tiling-head releases a tile at that location and on completion does not return to the origin but continues to release tiles till the sensor indicates a tile occupying the location immediately in front of it. At this point it returns to the origin and the next command is executed. Due to this, the execution of the program is now coupled with the developing configuration.

The interpreter  $\mathbf{i}_{state}$  is identical to  $\mathbf{i}_{coupled}$  except that it maintains a state variable that is incremented each time it detects a tile in front of it and shifts to execute the next command.

## Evolutionary setup

### The fitness function

A behavior evaluation function  $\mathbf{e} : \Phi_\zeta \rightarrow \mathbb{R}$  is defined as:

$$\mathbf{e}(\phi) = \sum_{i=1}^n (d(T_i, A) + d(T_i, B))$$

where  $\phi \in \Phi_\zeta$ ,  $T_i$  is the location of the  $i^{\text{th}}$  tile in  $\phi$ , and  $d(p, q)$  is the non-linearized version of the Manhattan distance between two points  $p$  and  $q$  in  $\zeta$  where  $d(p, q) = |x_p - x_q| +$

$|y_p - y_q|$  if  $|x_p - x_q| \leq M/2$ ; and  $d(p, q) = M/3 + |y_p - y_q|$  otherwise.

In words, the value  $\mathbf{e}(\phi)$  is the sum of the distances (as defined by  $d$ ) of every tile in a tile-configuration  $\phi$  from two pre-chosen points  $A$  and  $B$  in the workspace  $\zeta$ . The points chosen here are  $A = (1, M)$  and  $B = (M, M)$  where  $M$  is the length of each side of the workspace. These points correspond to the lower left and lower right corners of  $\zeta$  in Figure 1. The fitness of a configuration is related to the presence of specific features of the configuration. Configurations that have tiles concentrated close to  $X = 5$  axis, and away from the  $Y = M$  edge would tend to have higher values as compared to one where the tiles are randomly distributed around  $\zeta$ . Construction of these features require tight correlations between the locations of the tiles that are released, and the fitness drops rapidly with the lack of correlation.

## Algorithm

Each genotype is represented as a tape consisting of a string of integer values specifying a series of  $(x, y)$  coordinate pairs in the workspace  $\zeta$ . These are treated as being of fixed length defining  $n$  locations. So the genotype space  $\mathcal{G}$  consists of  $(M^2)^n$  programs where  $M$  is the length of a side of the workspace  $\zeta$ . Here  $M = 10$  and  $n = 20$ . The maximum number of tiles available for release is also equal to 20. The maximum value of  $\mathbf{e}$  for these parameter values is equal to 414.64.

The algorithm used here is similar to a canonical evolution algorithm having the structure

$$\dots G_i \rightarrow P_i \rightarrow P'_i \rightarrow G'_i \rightarrow G_{i+1} \rightarrow P_{i+1} \rightarrow P'_{i+1} \dots$$

where  $P_i$  represents the  $i^{\text{th}}$  generation of individuals with phenotypes on which selection acts, and  $G_i$  represents the gene pool of the generation.

The gene pool  $G_0$  is initialized with a random sampling of  $\mathcal{G}$ . The size of the gene pool is maintained constant at 30 throughout.

**Contingent development ( $G_i \rightarrow P_i$ ):** Each of the genotypes in this gene pool generates a maximum of  $N$  individuals by a developmental process involving the interpreter  $\mathbf{i}$ . Only a fraction  $k$  of these  $N$  is assumed to come to maturity and enter the reproductively viable population  $P_i$ . Here  $N = 10$  and  $k = 0.5$ . So, the population always consists of  $30 \times 5 = 150$  individuals. The developmental context i.e. the interpreter and the environmental noise model is assumed to be the same for the entire population and throughout the evolutionary process.

When the interpreter is  $\mathbf{i}_{coupled}$ , the first  $kN$  developed individuals are added to the population without any exclusion. However with the interpreter  $\mathbf{i}_{state}$ , whether an individual enters the population depends on the state accumulated during the developmental process of the individual. Here we pick a hand-designed value for this threshold to be 25% of the total number of tiles, so an individual enters the population

only when the accumulated state crosses this threshold. If no instance crosses this threshold after  $N - 1$  trials, the last individual always enters the population.

**Fitness assignment ( $P_i \rightarrow P'_i$ ):** Based on the behavior of the individuals as determined by  $\mathbf{e}$ , the population  $P_i$  is assumed to take on a reproductive viability in relative proportion to their behavior. This redistribution of  $P_i$  in terms of their relative fitness is represented as  $P'_i$ .

**Selection ( $P'_i \rightarrow G_i$ ):** In proportion to their relative fitness in  $P'_i$ , a subset of the individuals in the population and hence their genotypes  $G'_i$  are selected to be reproduced. Due to the fixed size of the gene-pool, 30 individuals in the population are selected for asexual reproduction in each generation. The selection is elitist with the default selection of the top 2 individuals with highest fitness.

**Variation ( $G'_i \rightarrow G'_{i+1}$ ):** With the exception of the genotypes corresponding to the elite individuals, the selected genotypes differ from the genotypes that constructed the parents. This difference is implemented explicitly with a variation operation to produce the gene pool of the next generation  $G_{i+1}$ , from  $G'_i$ . The variation is restricted to single-locus mutations of the parental genotype, where the probability of a mutation occurring is uniform over the entire parental genome. A mutation results in a random change in a location specified within a local neighborhood  $\mathcal{N}$  to the extent that it lies within  $\zeta$ . The neighborhood of a location  $(x_0, y_0)$  is given as  $\mathcal{N}_{(x_0, y_0)} = \{(x, y) : (|x - x_0| \leq r, |y - y_0| \leq r) \vee (|x - x_0| = 0, |y - y_0| \leq r)\}$  where  $r = 2$ .

## Results

The results from a representative run with the interpreter  $\mathbf{i}_{coupled}$  are shown in Figures 3 and 4. Firstly, we see that the fitness of the best individual does not change much throughout even in the first generation. However this has a pronounced effect on the fitness values in the population as the mean fitness can be seen to increase rapidly with the first 15 generations, but then remains flat after that. However the stable value of the mean fitness is significantly below the best fitness due to the large variation in fitness values for each genotype as can be seen in Figure 4.

In comparison, the fitness of the best individual and the mean fitness with  $\mathbf{i}_{state}$  (Figure 5) is of the same order as  $\mathbf{i}_{coupled}$ . However, there is a significant difference in the interval lengths of the individuals entering the population with  $\mathbf{i}_{state}$  (Figure 6). As the best and mean fitness are of the same order as  $\mathbf{i}_{coupled}$ , this is due to an increase in the value of  $e_{min}$  of the genotypes in the population though without an increase in  $e_{max}$ . This reduction in the interval length can be seen as the downward shift in the scatter plot as compared to that of  $\mathbf{i}_{coupled}$ .

The higher specificity with  $\mathbf{i}_{state}$  can be attributed to the manner in which the maintained state is used to selectively exclude individuals from the population based on the contingencies of their particular ontogenies.

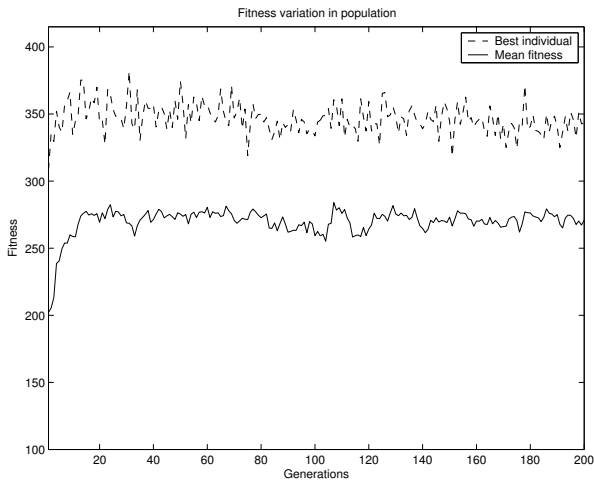


Figure 3: Fitness variation with  $i_{coupled}$

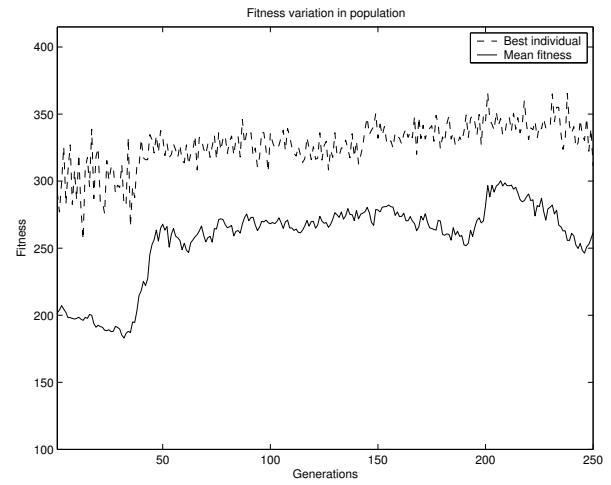


Figure 5: Fitness variation with  $i_{state}$

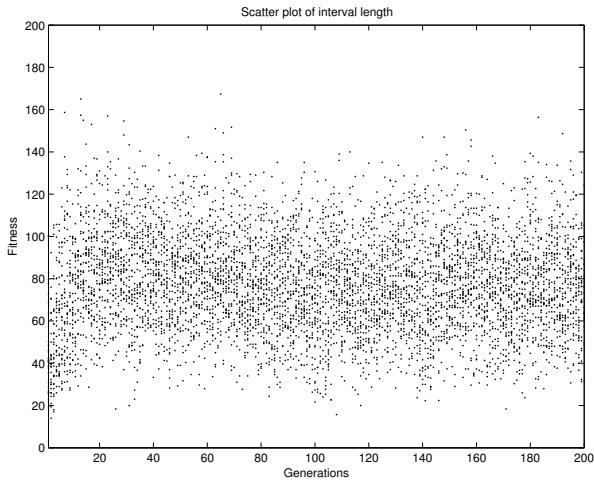


Figure 4: Scatter plot of interval length with  $i_{coupled}$

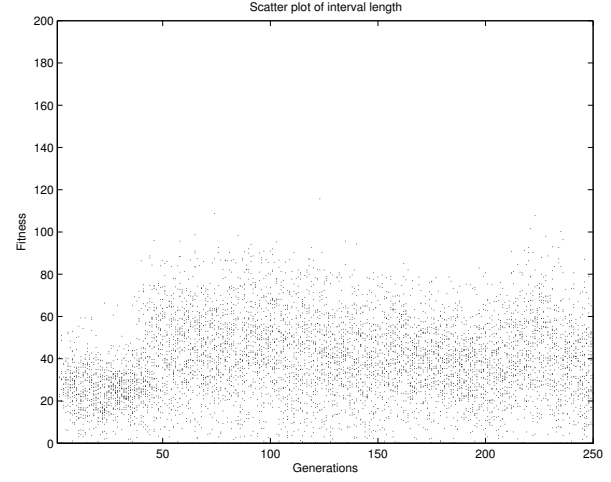


Figure 6: Scatter plot of interval length with  $i_{state}$

The rationale is that at different stages in development there may exist low-dimensional measurable properties and events that provide information about the global state. So developmental elements whose behavior is contingent on the “measurements” of this low dimensional information could influence subsequent states in the ontogenic trajectory. However, developmental interactions are hidden from the selective pressure acting on the population. So on evolutionary time scales these contingent responses can be “calibrated” by evolution such that they are related to the fitness of the outcomes.

In the case of the tiling-machine, a sensor response of  $\mathbf{1}$  at the head occurring immediately after release indicates that the tile was at the very location it was released at. On the other hand a sensor response of  $\mathbf{0}$  after releasing the tile indicates uncertainty as it provides no information about where the tile would eventually come to rest. So a larger number

of  $\mathbf{1}$ s would indicate greater certainty about the outcomes of the tile release operations though it says nothing about the fitness of the outcome. As can be seen in Figure 7 that plots the accept rate i.e. number of individuals accepted relative to those that were obtained, the accept rate is about 50% as compared to the 100% in the case of  $i_{coupled}$ .

The existence of such low-dimensional measurable properties containing global-state information is tested empirically by determining whether evolution can find a way to use the additional state maintained by  $i_{state}$  to influence the outcome distribution. This “calibration” to exclude individuals that correspond to low-fitness individuals is clearly observable in this case.

## Discussion

The tiling machine system described here is primarily used here as a way to provide a transparent and qualitative

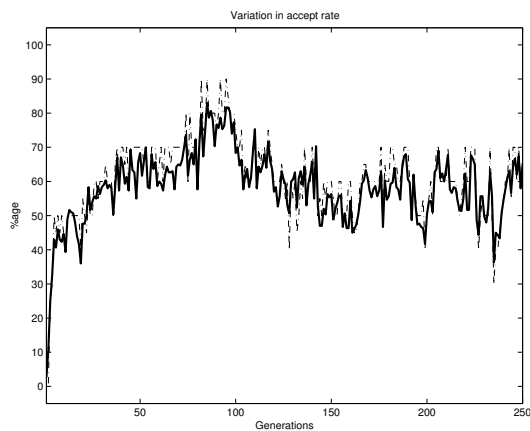


Figure 7: Accept rate

demonstration of the issues to be addressed toward a unified functional-embodiment approach to the evolution of complex real-world systems.

The fundamental difference that arises with the presence of non-genetic variation in the population is that selection of genotypes is now an ambiguous process as the observed fitness differences in the individuals in the population may not correspond to heritable variation. More importantly, this variation is not due to lack of precision in evaluating the fitness but involving systemic variations in the structure and behavior between the different developmental outcomes. So there is no “true” phenotype associated with a genotype with the other phenotypic variants being the effect of noise. As a result the key problems are related to how order and structure can be extracted from the high variability involved. Here we have used a simple experiment to demonstrate the critical role that developmental substrates can play in this context by enabling the evolution of specificity. A more comprehensive analysis of this issue is presented in (Viswanathan and Pollack, 2004).

By incorporating interactional specificity into construction but without reference to a “goal” structure and showing the relation of embodiment processes to the behavior rather than structure of the outcomes, this paper presents an incremental step toward bridging the gap between the functional and embodiment perspectives.

## Conclusion

This paper identifies a class of problems arising due to the idealizations in functional and embodiment perspectives of evolutionary development. Toward addressing this issues, a preliminary abstraction for the analysis of evolvable developmental substrates is described and demonstrated using a simple toy-system. Experiments with this system suggest that the developmental substrate could exert adaptive influences on the ontogeny at the interactional level.

## References

- Brooks, R. (1991). Intelligence without reason. In Myopoulos, J. and Reiter, R., editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.
- Brooks, R. (1992). Artificial life and real robots. In Varela, F. and Bourgine, P., editors, *Toward a practice of autonomous systems: Proceedings of the First European Conference on Artificial Life*, pages 3–10. MIT Press.
- Crane, H. R. (1950). Principles and problems of biological growth. *The Scientific Monthly*, 70(6):376 – 389.
- Funes, P. and Pollack, J. B. (1998). Evolutionary body building: Adaptive physical designs for robots. *Artificial Life*, 4(4):337–357.
- Harvey, I., Husbands, P., Cliff, D., Thompson, A., and Jakobi, N. (1997). Evolutionary robotics: the sussex approach. In *Robotics and Autonomous Systems*.
- Hornby, G. and Pollack, J. (2002). Creating high-level components with a generative representation for body-brain evolution. *Artificial Life*.
- Lipson, H. and Pollack, J. B. (2000). Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978.
- Pfeifer, R. (1996). Building fungus eaters: Design principles of autonomous agents. In *From Animals to Animats 4: Fourth International Conference on Simulation of Adaptive Behavior*, pages 3–12. MIT Press/Bradford.
- Pollack, J. B., Lipson, H., Funes, P., Ficici, S. G., and Hornby, G. (1999). Coevolutionary robotics. In Koza, J. R., Stoica, A., Keymeulen, D., and Lohn, J., editors, *The First NASA/DoD Workshop on Evolvable Hardware*. IEEE Press.
- Simon, H. A. (1962). The architecture of complexity. *Proceedings of the American Philosophical Society*, 106:467–482.
- Sims, K. (1994). Evolving 3D morphology and behavior by competition. In Brooks, R. and Maes, P., editors, *Artificial Life IV*, pages 28–39. MIT Press.
- Stanley, K. O. and Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artificial Life*, 9:93–130.
- Viswanathan, S. and Pollack, J. (2004). On the evolvability of replication fidelity in stochastic construction. Technical Report CS-04-248, Brandeis University.