# Self-Replication, Evolvability and Asynchronicity in Stochastic Worlds

Chrystopher L. Nehaniv

Adaptive Systems, Algorithms, and BioComputation Research Groups
School of Computer Science & Science and Technology Research Institute
University of Hertfordshire
Hatfield, Hertfordshire AL10 9AB
United Kingdom

C.L.Nehaniv@herts.ac.uk

*In Memory of John Maynard Smith and Victor Varshavsky*

**Abstract.** We consider temporal aspects of self-replication and evolvability – in particular, the massively asynchronous parallel and distributed nature of living systems. Formal views of self-reproduction and time are surveyed, and a general asynchronization construction for automata networks is presented. Evolution and evolvability are distinguished, and the evolvability characteristics of natural and artificial examples are overviewed. Minimal implemented evolvable systems achieving (1) asynchronous self-replication and evolution, as well as (2) proto-cultural transmission and evolution, are presented and analyzed for evolvability. Developmental genetic regulatory networks (DGRNs) are suggested as a novel paradigm for massive asynchronous computation and evolvability. An appendix classifies modes of life (with different degrees of aliveness) for natural and artificial living systems and possible transitions between them.

## 1 Models of Time: Logical vs. Physical Time

We consider time in discrete dynamical systems. St. Augustine considered time as something intuitively graspable yet ineffable. Varshavsky distinguished two kinds of time: Time as a *logical variable* in a system defined by events vs. time as an independent *physical variable* [96], and studied self-timing and asynchrony theory for computing devices as the problem of reconciling the two types of time via design of system timing for the appropriate functioning asynchronous devices interacting with external environments.

For a single observer or location, we can consider three main views of the (logical) time:

O. B. Lupanov et al. (Eds.): SAGA 2005, LNCS 3777, pp. 126–169, 2005.

[This version includes minor corrections to the published text.]

### 1.1 Partial Orders as Models of Time

Aristotle considered events in time via ordering related to casuality (and motion), and time as defined by differences between states before and after (thus change is required for the passage of time).

### 1.2 Time as a Random Variable

Another view is to regard logical events, such as a discrete event clock-tick, as embedded in physical time but where a random variable takes values *event* or *no event* according to some distribution at successive discrete moments of physical time. (Instead of just one type of *event* more generally different particular events might be generated.) Here the passage of logical time, if used to increment a measuring counter, is monotonically but not deterministically related to the passage of physical time.

### 1.3 Algebras of Time: Semigroups as Models of Time

Following J. L. Rhodes (who refers to Aristotle), we can describe time algebraically. If $\alpha$, $\beta$, and $\gamma$ are each sequences of events in time, and the composite sequence $\beta$ then $\gamma$ is preceded by $\alpha$, this is exactly in the same as when $\beta$ follows $\alpha$ and after both $\gamma$ occurs. That is, the *associative law*

$$(\alpha\beta)\gamma = \alpha(\beta\gamma)$$

is a grammatical statement about sequences of events in time. The study of associative structures (semigroups) is thus the study of models of time.

As a simple application, we have algebraically classified all models of time allowing for only a single repeated event (or “clock-tick”).[1]

Semigroups are intimately connected with deterministic automata, as sequences of inputs induce mappings of the set of states of the automaton to itself; these induced mappings thus comprise a semigroup (under the associative operation of function composition) which serves as a model of time in the automaton.

To pass to a nondeterministic or probabilistic automaton, there are several methods. A very general one, related to the construction of minimal automata, applies to the more general case of observations or measurements of any phenomenon at all. Observations of a given stochastic phenomenon can be treated via Crutchfield’s $\epsilon$*-machines*: from observations of an, in general stochastic, process one constructs a deterministic automaton in which transitions are single observations and in which states are equivalence classes of past histories for each member of which the probability distribution over the future histories is identical. In other words, in a given state the future is conditionally independent of the past [15]. The semigroup of the $\epsilon$-machine then serves as an algebraic invariant and model of the temporal dynamics of the given phenomenon.

[1] The possible single event models are cyclic groups, the positive natural numbers under addition, and thresholded cyclic groups – in which the event can be repeated some number of times whereupon one enters a cyclic group [57].

## 2 Evolution and Evolvability

Evolution viewed as stochastic synchronous or asynchronous algorithm or temporal process is described here. Evolvability describes the capacity to which a particular evolutionary process is successful in generating adaptive individuals and will be discussed in detail later. After defining Darwinian evolution, we survey non-biological examples and the other evolution-like phenomena. Evolvability is then discussed in detail for these examples.

### 2.1 Definition of Darwinian Evolution

**Evolution** *is any* **dynamical population process** *[17, 16] with the following characteristics, which one can regard as the semi-formal* **Darwin-Wallace axioms***:*

*(1)* **Heritability***: Individuals have inherited information or material (genotype) from parent(s) that makes them similar to their parent(s) in some traits.*
*(2)* **Variability***: Offspring may differ from their parents in their heritable material (genotype) and in other respects (phenotype).*
*(3)* **Differential Reproductive Success [selection]***: depending on phenotype, which must depend at least in part on inherited traits, some individuals are more likely to have any (or more) offspring than others.*
*(4)* **Finite Resources and Turn-over of Generations***: Lifespans of individuals are finite and the existence of only a limited number of individuals can be supported in the population at any moment.*

The above axioms yield a creative engine via a "*struggle for existence*" driving "*descent with modification*". Persistence and increase in distribution of heritable successful traits follow by (1) and (3), and creativity arises via (2). Competition for existence is due to (4). Note that a presupposition of these axioms is that the population consists of well-defined individuals.

### 2.2 Stochasticity of Evolution

Stochasticity impinges on evolution usually (1) via the mechanism of genotypic variability, whereby inherited information is perturbed, but also (2) in phenotypic variability whereby the environment or constrained aspects of development lead to differences between parent and progeny. Differential reproductive success (3) refers to the probability of success at producing progeny depending on inherited information and is therefore generally modeled as stochastic in nature.[2]

Evolution can thus be regarded as a very general class of stochastic algorithm with many instances occurring in nature, culture, and artificial systems.

[2] Nevertheless, evolution is also possible in completely deterministic systems, e.g. the synchronous evoloop system [77] (see below), or any non-interactive genetic algorithm running with a given 'seed' for generating random numbers.

## 2.3 Instances of Evolution *in Silico*

**Genetic Algorithms and Evolutionary Computation** Genetic Algorithms (GAs) and allied methods are population processes for artificial evolution in computers and have been introduce in many variants: *genetic algorithms* [27], *evolutionary strategies* [70, 80], *evolutionary algorithms* [24], and others. The "vanilla" genetic algorithm in the style of Holland [27] is described here:

0. Create a population of fixed finite size of fixed-length bit-strings encoding candidate solutions to an optimization problem (initialized randomly or with domain knowledge)
1. Evaluate each against an objective function ("fitness function")
2. Copy individuals that do better with higher probability into next generation (a new population, with same population size) [selection]
3. Apply variability operators: mutation random bit-flips, crossover: recombination between two individuals by swapping the substrings after some randomly selected point along the strings, and others.
4. Iterate 1-3 until satisfied.

**Genetic Programming: variant of GA** Genetic Programming (GP) is a variant of GAs introduced in the early 1990s by John Koza [30]. It has the following structure:
0. Do GA, but on populations of programs, not bit-strings (e.g. Lisp S-expressions, or parse trees in any programming language). Individuals are syntactically correct programs over some chosen set of basic operations, and terminals (constants and variables).
1. Behaviour or output of each program in population is evaluated against an objective function ("fitness function")
2. Copy individuals that do better with higher probability into next generation (a new population, with same population size) [selection]
3. Apply variability operators respecting syntax: mutation replaces a subtree by random one (of the same type); crossover: exchange subtrees (of the same type) at random nodes between two individuals. New individuals are syntactically correct since the operators respect node typing.

Later variants of GP also introduced explicit support for modularity, named functions (so-called automatically defined functions (ADFs)) that can be called by the main result producing branch of the program.

**Digital Organisms** Digital organisms were introduced by tropical evolutionary biologist and computer scientist T. S. Ray around 1989. Individuals in a finite computer memory are self-replicating programs running on a Darwinian operating systems (one in which mutations in data and flaws in computational operations occur with certain low probabilities). There is no objective function, so we have an instance of *natural selection*. The motivation is not optimization but artificial life as a generalization of biology. Several systems for the evolution of digital organisms have been developed: `Tierra` [69], the first one, gives rise to rich ecologies (parasites, obligatorily social hyperparasites, etc.). Insights that make it evolvable (as compared to random mutation of computer code) were inspired by biology and include (1) template recognition and matching (recognition based on "shape"); (2) all strings are syntactically correct (in Tierra, assembler-like programs), and (3) small language size - no numerical constants are permitted in statements (but must be constructed in an organism's digital processor if needed). Space in computer memory and processor cycles are the fundamental resources for digital organisms; these and interaction amongst digital organisms determines their reproductive success in an emergent manner. `NetTierra` is an internet wide version with multithreading (an analogue of multicellularity), sensing by digital organisms of other sites over the network, and migration between computers. The `Avida` system added CPU cycle rewards for some computations and is currently most widely used [2], especially as a model for bacterial evolution and an experimental test-bed for population genetics theory. `Physis` [23] is new system for studying evolvability of digital organisms in which organisms carry not only the code for their self-replication, but also code specifying the processor that will run it and, moreover, code specifying the language they will run on it: evolvable processors. This latter system allows the study of the evolvability of self-replication, including phenomena analogous to evolution of the genetic code for protein biosynthesis via translation to amino acid sequences from sequences of codons in very long oligonucleotides (DNA/RNA).

## 3 Not-Quite-Darwinian Evolution

Several cases of what looks like evolution (and is often called evolution) fail to meet the Darwinian axioms. Generally, in such cases, there are dynamical similarities, but the problem is that one cannot identify well-defined individuals. An analogue of producing progeny in such cases is *persistence* [52], usually eventually with modification (and hence variability).

### 3.1 Software Evolution

In Software Engineering, the costs of so-called 'software maintenance' and 'software evolution', i.e. costs of modifying and adapting already released software, amount to billions of dollars annually (50-95% of all software costs [83, 35]). Software is static, fragile and inflexible (except where adaptation need has been foreseen), but its context and environments of use change, hence requirements change. Software evolution has been characterized as managing change – see the work of Lehman, Goguen, and Berners-Lee (e.g. [35, 25, 26, 8]). Persistence and re-use of software is an analogue to heritability [52]. If software code is regarded as heritable information, the severe problem of requirements change shows the need for software that possesses *phenotypic versatility* and *robustness to perturbation*, both of which are related to evolvability.

Software growth has been studied as a dynamical system with system-level, positive and negative growth laws [103]. There are no clear individuals, no population. But there is persistence and growth, and descent with modification. Are there principles in common with those of biological evolvability? The answer seems to be yes, but they are not well-understood yet. Any software carries with it an unbounded number of hidden assumptions, which are progressively violated as time passes and context of use – and hence requirements – changes [35]. A design principle similar to biological evolvability:

attempt to be future-proof, robust to likely sources of change (see [8] on future-proofing and world wide web data and mark-up languages).[3]

### 3.2 Cultural Evolution

Other examples of evolution without readily identifiable individuals in populations are the evolution of artifacts, and the evolution of behavior or cultural (memetic) evolution.

## 4 Evolvability

The evolution of life on earth has undergone several major transitions. Major transitions in evolution are studied in [11, 45]: Free Replicators to replicators in compartments; RNA as gene/enzyme to DNA and protein (genetic code); prokaryotes to eukaryotes; asexual clones to sexual populations; protists to differentiated multicellular life (esp. [11, 47]); solitary individuals to colonies with non-reproductive castes [45].

All of them involve transitions in the way information is used and most of them involve the advent of new types of individuality and thus new units of selection in populations of these new individuals.

Nothing like the complexity and creative power of organic evolution has been realized in artificially constructed evolutionary systems. Why is this the case? Computer scientists using evolutionary computation techniques quickly discovered that in some cases evolution was better able to find solutions than in others. Sometimes evolution completely failed as an optimization method, other times it worked well. Biologists had tacitly assumed that evolution by itself was sufficient to generate open-ended adaptivity and complexity of the kind they observed in nature (e.g. flowering plans, animals with complex body plans, etc.). But the frustration of computer scientists in some cases showed clearly that some systems were obviously more evolvable than others.

### 4.1 Krohn-Rhodes Complexity and Open-Ended Evolution

This leads to a constructive challenge problem.

**Open Problem 1. (Open-Ended Evolution)** Build a system that exhibits open-ended evolution. One in which complexity can grow arbitrarily large and new innovation and complex traits continue to arise.

[3] The problem with being "future-proof" is that evolution by itself is a historical process that cannot predict anything about the future. In biology, robustness to likely sources of change appears to be achieved via *lineage selection*, i.e. lineages robust to the kind of change that has historically occurred are more likely to continue than others when changes of the same type reoccur in the future. In software, human design as well as such lineage selection may operate. See also the discussion in the sections of this paper on GP code bloat and on the evolution of evolvability.

Krohn-Rhodes complexity in algebraic automata theory using semigroups as models of time (or, e.g. Kolmogorov complexity) can be used to formalize the notion of unbounded complexity growth, and explicit bounds on complexity increase in the course of smooth evolution can be computed [60]. Duplication-and-divergence is one generic method of maximizing jumps in complexity [62].

### 4.2 Origin vs. Fate of Variation

Most evolutionary theory (e.g. nearly all of population genetics) has been concerned with the fate rather than origin of variation [101]. Variability is the only source of creativity in the evolution axioms, and its generation must therefore be one of the keys to evolvability.

### 4.3 Definition of Evolvability

Evolvability has been characterized in various ways in the literature:

- "the ability of a population to produce variants fitter than any yet existing" (Altenberg [5])
- "genome's ability to produce adaptive variants when acted on by the genetic system" (Wagner & Altenberg [101])
- "the capacity to generate heritable phenotypic variation" (Kirschner & Gerhart [29])
- characterized by evolutionary watersheds opening the floodgates of evolution, such as with the advent of segmentation and body plans (Dawkins [20])

A synthetic definition is formulated here:

**Definition. Evolvability** *is the capacity of a population to generate adaptive heritable genotypic and phenotypic variation.*

In this definition, "**adaptive**" is understood as *fitter than any currently existing*.[4]

### 4.4 Genetic Algorithms and Evolutionary Computation: Evolvability Issues

Choice of *encoding* is a crucial issue for evolutionary computation: "The Representation Problem". Encoding determines the genotype (e.g. bit-string) to phenotype (fitness evaluation) mapping (Genotype-Phenotype Map).

[4] This notion of evolutionary adaptivity is similar in its sense to that used in Altenberg's definition above [5], but is more specific in that it replaces "fitter than any *yet* existing" by "fitter than any *currently* existing". The reason for this is that fitness is a spatio-temporally *local* notion depends of the current organism-environment interactions and niches which are of course subject in general to temporal variation over generations.

The production of fitter individuals might first proceed via *neutral evolution*, i.e. the production of new individuals with different genotypes and of equal fitness to those existing; this is known to increase evolvability in many examples (cf. [28, 93].)

### Genotype-Phenotype Relation

> “The genotype-phenotype map is the common theme underlying such varied biological phenomena as genetic canalization, developmental constraints, biological versatility, developmental dissociability, morphological integration, and many more” - G. P. Wagner & L. Altenberg [101]

Variability operators determine the topology (neighborhood relations of genotypes) of the *fitness landscape* (S. Wright 1932 [107]), mapping genotype (or genotype and phenotype via environment interaction) to probability of reproductive success. Smoothness of the objective function on this landscape determines how well GAs can do their stochastic hill-climbing. If there are deep broad valleys between fitness peaks (local optima) that can’t be traversed quickly enough, the system is not evolvable. Conversely, uphill paths reachable by a single step from local optima help make a landscape evolution-friendly.

To improve evolvability, the evolutionary strategies of Rechenberg and Schwefel [70, 80] introduce the heritability of locus specific mutation parameters (for the variance of noise applied to numerical parameters under optimization).

*Extradimensional bypass* [14] is the adding of dimensions to the genetic ‘search’ space (e.g. by an insertion mutation or by duplication of a gene), in higher dimensional fitness landscapes, local optima often become saddle points; this is observed in protein evolution, and is related to neutral networks and robustness (via mutational buffering). Sometimes it has been used in evolutionary computation, e.g. via growth in genome size or duplication of all or part of the genome, to achieve improved evolutionary performance.

## 4.5 Genetic Programming and General Evolvability Issues

In GP, an important phenomenon is code bloat: for robustness to crossover, size of programs increases uncontrollably. They are full of junk in order to withstand crossover with lower chance of distribution. Making multiple crossover occurrences more likely for large trees according to their size eliminates this trend [91].

This is a particular instance of a general principal in the evolution of evolvability: *Evolution favors lineages with robustness to disruption from the variability operators experienced by the evolving population.* See [66] for a related study on linkage and crossover, and [92] for the neutral evolution of mutational robustness.

Modularity: Automatically Defined Functions (ADFs) [31] are functional modules that can be called from various locations in a program. Using these can measurably increase evolvability [90]. This is related to analogous principles in software engineering for evolvability: code factoring, appropriate modularity, and re-use (e.g. [65, 52, 83]).

## 4.6 Properties and Mechanisms of Evolvability

What makes an instance of the stochastic algorithm, evolution, evolvable? A list of properties and mechanisms that seem closely related to evolvability is presented here.

In many cases it is unclear whether we are examining a prerequisite for, or a consequence of, evolvability, or possibly both (via the circular casuality of the dynamical evolutionary process), or perhaps an incidental property.

1. Developmental Plasticity: Universal responsiveness to interaction with the environment, an incessant, continual coupling throughout life. (lifelong viability; multiple cell types; complex life cycles; multiple developmental pathways/behaviours/morphs; continual self-creation and maintenance in interaction with environment/others). This property is almost unknown in artificial systems, standard population genetics models, the 'new synthesis', unimodal evolutionary models; but see West-Eberhard [104] and also Varela [94].
2. Flexibility/Rigidity of Genotype-Phenotype Relation. Robustness (to Heritable and to Developmental/ Environmental Perturbations)
3. Duplication-Divergence: From one, many! (cell types, castes, genetic regulatory networks (GRNs), segmentation, generic complexity increase)
4. Differentiation, Local Adaptation and Control
5. Appropriate Modularity, Compartmentation. Potential to Combine Lower Level Units: one from many!
6. Symbiogenesis
7. New Individuality (e.g. Multicellularity, Compartmentation; Linkage)
8. Use of Signaling, Switches, Signal Transduction, & Feedback Control
9. Employment of Evolutionary Dynamics (within individuals!)
10. Redundancy
11. Extradimensional Bypass

### 4.7 Duplication and Divergence.

Gene duplication is remarkably frequent and important in biological evolution [63], and subject to complex evolutionary dynamics [39]. The creation of a full or partial extra copy of a gene (or other component) frees one copy or both copies to specialize functionally, or one copy to acquire a new function. Duplication and divergence in biological evolution [63] is thus a generic mechanism for the generation of variability, of great potential creative power.

Duplication and divergence (Figure 1) is also exemplified by division of labor among cells or tissue types in a body, or castes in a social insect colony. *In differentiated multicellularity, growth via cell division together with specialization into cell types (e.g. into soma and germ lines) provides an opportunity and mechanism to exploit asynchronous parallel processing by closely related entities to achieve adaptation at a higher level of individuality.*

Complexity increase via duplication and divergence, e.g. increase in number and role of cell types [9], or acquiring genomes [43] (which does not involve duplication and divergence) can apparently realize known, sharp theoretical bounds on the evolution of biological complexity [60].

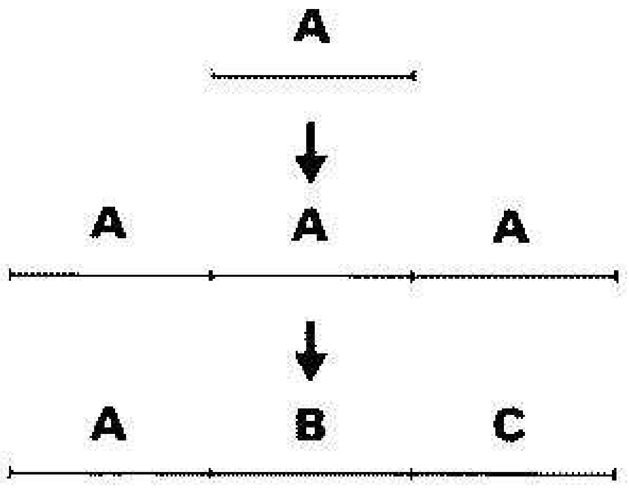


**Fig. 1.** Duplication-Divergence: A Generic Path toward Complexity Increase and Evolvability. (after J. Maynard Smith [44])

**Differentiation** Differentiation of multiple copies of the same entity as in differentiated multicellular involves the following properties:

- Multiple copies of regulatory mechanism in similar units (e.g. genetic regulatory networks, cells, individuals, etc.)
- Local state
- State inherited by lineage (e.g. Cell types, growth and morphogenesis, epigenetics via methylation, etc.)
- Local adaptation to local conditions
- Long- and medium- distance interactions
- Growth from single unit to a differentiated many, with changing topology
- Division of labour

## 5 Self-Production and Reproduction

How is it possible for a mechanistic system to produce something as complex or even more complex than itself? This problem motivated von Neumann to study the physical and logical basis of self-reproduction using automata models. Von Neumann considered automata capable of (1) examining and copying any pattern or specimen given to them, or of (2) production of any object starting from a logical description.[5] Either approach

[5] See especially notes of von Neumann's fifth lecture "Re-Evaluation of the Problems of Complicated Automata - Problems of Hierarchy and Evolution" in Part I of [99] delivered in December 1949 (and edited and reconstructed by A. W. Burks). In both cases the word "any"

leads to a solution to von Neumann's problem: in the first method, present the automaton with an entity as or more complex than itself; in the second, present it with a logical description of one. Of course for this to work, it is necessary to construct such *universal constructing* automata with these capabilities or to demonstrate their existence.[6]

These two approaches lead to solutions of the problem of self-reproduction by *self-examination* vs. *heritable encoded information* respectively. One presents the universal constructing automaton with itself (or a copy of itself), or with a logical description of itself, respectively. In the latter case, a copying component of the constructing automaton can be used to copy the logical description (regarded as part of the entity), which thus becomes *heritable* genetic information. Von Neumann showed how to construct such an automaton in a synchronous cellular automata network using the second method [99, Part II].

Mutations or errors in the construction process could lead to lethal or non-lethal variant copies and hence provide the variability required for evolution to act. Conceivably, this could therefore lead to the evolution of more and more complex automata. Although von Neumann considered this possibility, so far no one has been able to shown in detail how it could be realized.[7]

It is remarkable that von Neumann's solution used genetic, inherited information in two roles: (1) blindly copied and (2) executed, before the structure of the heritable genetic material in life on earth was uncovered by Watson and Crick's 1953 detailed description of the structure of DNA revealing its essentially *digital* nature with similar dual roles [102]. Thus, von Neumann's work on his automata models even anticipated the important transcription ("blind copying") and translation ("executability") properties of genetic material found for DNA, with the former realized by complementary pairing of bases and the latter via template matching and the genetic code sequentially mapping codons (triplets of "letters" of DNA) to the amino acids in proteins (along with numerous regulatory intricacies).

From the beginnings of the study of self-reproduction in artificial systems initiated by von Neumann already in 1948, the primary formal model has been synchronous cellular automata in which configurations develop that eventually may include an unbounded number of copies of the original. The models constructed by von Neumman and his successors have amply demonstrated that self-reproduction is indeed possible in artificial systems.

The different possibilities for achieving self-reproduction have implications for our understanding of the origin of life, the nature of organic life, and for the possibilities of life as it may exist elsewhere in the universe. Szathmáry [84] offers a classification of replicators applicable to natural and artificial systems along the dimensions of the replication process (*holistic* vs. *modular*, and *genotypic* vs. *phenotypic* (the latter is defined by non-modular copying of functionality)) and of variability (*limited* vs. *unlimited*

must be taken as having scope over a particular very large class of bounded structures whose existence is possible in the ambient environment.

[6] Portions of this section are based on the author's paper [55].

[7] To demonstrate this, a suitable rigorous complexity measure would of course be a pre-requisite (cf. [62]).

*heredity*, where the latter requires that the number of possible variants be much larger than the number of individuals in the population).

Self-reproduction is of course a prerequisite for any independent evolutionary process. Sending information, instructions on how to build copies of desired structures using local materials, into an environment rather than sending all necessary materials into that environment represents more economical methods of space exploration and colonization. See the NASA report edited by Freitas and Gilbreath (1980) for further potential examples and applications of self-reproduction to space science, e.g. self-replicating and self-maintaining lunar factories.

### 5.1 Self-Replication and Time

*Nature abounds with asynchrony.* Cells in a multicellular organism or organelles or molecules within a cell apparently have no access to a central clock signal. Can von Neumann's problem still be solved without synchrony? Might the restriction to synchronous update be relaxed? In building an artificial self-reproducing entity is it really necessary to have a single global synchronization signal that reaches all parts of the entity simultaneously (or at least within a well-defined tolerance)? If local parts of the configuration are ready to change their state, is it realistic and practical to assume that they must wait until all other parts of the cellular space are also ready to update their states?

We can indeed free all cellular automata models of self-reproduction as well as all cellular automata models of evolution, universal computation, and universal construction from the need for synchronous update ([55, 53], and below). This is accomplished by an elegant simple mechanism that allows one to construct an asynchronous automata network that is capable of emulating the behavior of a given synchronous automata network. State updates in the asynchronous model may be produced by practically any asynchronous update mechanism whatsoever[8] (e.g. updates may be sequential, occur randomly – locally distributed according a probability distribution, be partially simultaneous, etc., or even synchronous). The result for cellular automata is a special case of a more general theorem for automata networks with inputs due to the author (Theorem 1 below, [56]).

We describe below the construction for making any automata network's computation asynchronously realizable, give examples that illustrate how the use of "local time" frees cellular automata networks from the need for global synchronization, and display asynchronous examples of self-reproduction and evolution in cellular automata in the context of discussing evolvability in natural and artificial systems.

### 5.2 Models of Self-Reproduction

Von Neumann's original constructive demonstration (begun in the 1940s and completed by Burks in the 1960s) of self-reproduction of a configuration of states in the cellular

[8] The only essential restriction is that each local automaton is updated an unbounded number of times, and a given node from the viewpoint of another cannot have been updated infinitely often in the past.

automata network has the properties that the self-reproducing configuration is capable of universal computation (in Turing's sense) and of universal construction – loosely speaking, the ability to fill any compact area in the cellular space with any desired pattern. These properties were included in addition to the ability of the replicator to make a copy of itself, and could also be used to support this ability. Namely, *universal construction* (as the ability to fill any compact region of the cellular space with arbitrary configurations) guarantees that a copy of the self (including its 'instruction tape' which is present in many examples) can be constructed. However, von Neumann's design of a self-reproducing universal computer and constructor was infeasibly large and has never been fully implemented and executed through a reproduction cycle on a computational device.

Langton's (1984) definition of self-reproduction requires that a copy is constructed but realizes neither universal computation nor universal construction [33]. Langton implemented and studied the first example of feasible self-reproduction in cellular automata, using an 8-state cellular automaton with an initial configuration of 86 cells, that produces a first offspring after 151 time steps and then proceeds to fill up available space with copies. To avoid trivialities while avoiding the complexity of von Neumann's model, *Langton's criterion* [33, 34] was proposed as a necessary condition on self-reproduction and requires that information is treated in the two ways identified above: as instructions that are *executed* ('translation') and as data which are blindly *copied* ('transcription'). These properties are also present in and abstracted from von Neumann's and later Codd's examples [13], and were by that time also known to be characteristic of biological self-reproduction. Encoding of heritable information in the shape of a configuration or using self-inspection represents another feasible mode of encoding heritable variation in self-reproduction (cf. [32, 69, 50, 54]). Subsequent examples of Byl [12] and Reggia et al. (e.g., [71, 38]) simplified the self-replicating loop of Langton toward minimality, with fewer states, simpler transition rules, or less cells in the initial configuration. In some cases the simplifications are so severe that it is debatable whether *nontrivial* self-reproduction has been achieved (e.g. according to Langton's criterion).

Subsequently, various researchers kept Langton's requirements for self-reproduction, but have added more and more computational power to the relatively small self-reproducing cellular automata configurations (in comparison to von Neumann's solution). These trends are surveyed by Lohn [37], who also describes the evolution of cellular automata rules that support self-reproduction (see also [38]). An annotated bibliography with some links to various relevant on-line resources can be found at Moshe Sipper's Artificial Self-Replication page [82].

H. Sayama [76, 77] has constructed variants of the self-reproducing Langton loop which exhibit self-dissolution once they can no longer reproduce, thus freeing up space for reuse by progeny, and most interestingly, another similar variant called "evoloop" which exhibits heritable variability in loop size and is subject to evolution via interaction among descendants of a common ancestor acting as a selective force ([75, 77], and below). Heritability, variability, and turn-over of generations with differential survival in an environment with limited resources are present in his evoloop when run in finite spaces. Thus evoloop appears to be the first convincing example of an evolutionary process occurring in cellular automata.

## 5.3 Self-Reproduction, Individuality, and the Heritability of Fitness

What constitutes self-reproduction?

The definition is not uncontroversial. We have already mentioned that von Neumann included universal computation and universal construction in order to exclude trivialities, such as the simple example of spreading activation. Langton abstracted the properties of inherited information being both copied and executed.

E. F. Moore [49] defines a configuration $C$ to be *capable of self-reproducing* $n$ *offspring by time* $t$ if starting from the initial conditions of the entire cellular space at time $t = 0$ such that the set of all non-quiescent cells of the space is an array whose configuration is a copy of $C$ there is a time $t' > t$ such that at time $t'$ the set of all non-quiescent cells will then be contained in an array whose configuration includes at least $n$ copies of $C$.

Lohn and Reggia [38] give the following definition:

> "A configuration $C$ is self-replicating if the following criteria are met. First, $C$ is a structure comprised of more than one non-quiescent cell and changes its shape during its self-replication process. Second, replicants of $C$, possibly translated and/or rotated, are created in neighbor-adjacent cells by the structure. Third, there must exist a time $t$ such that $C$ can produce $i$ or more replicants, for any positive integer $i$, for infinite cellular spaces (Moore's criterion). Fourth, if the self-replication begins at time $t$, there exists a time $t+\Delta t$ (for finite $\Delta t > 1$) such that the first replicant becomes isolated from the parent structure."

The issue of exactness of the copy is problematic since it is not desirable to exclude the possibility of variability. Variability among offspring is certainly present in biological systems, and, as Darwin showed us, is necessary for evolvability. Vitányi [97] introduced sexual reproduction in cellular automata and Sayama [76], mentioned above, has demonstrated variability and (deterministic) evolution occurring in cellular automata.

A discussion of the difficulties in formulating a rigorous definition of self-replicating or self-reproduction is given by Nehaniv and Dautenhahn [58], who point out that even in accepted cellular automata models of self-reproduction there are rarely two copies of the original configuration present at exactly at the same time when reproduction is generally accepted to have occurred (e.g. in the von Neumann or Langton models), and it is certainly not the case when the first offspring has been produced. The various copies of the configuration may be at different stages in their "lifecycles" and not have exactly the same configuration of states. They suggest looser criteria on identity of copies to allow 'species' of non-exact copies to be acknowledged as offspring, and also loosen the restriction on the presence of copies all at the same time (e.g., offspring that have to grow into adults are still regarded as offspring even though they are never in exactly the same state of development as the parent). *Adequate formal definitions of "member of the same species" and of "individual" are still lacking in the sciences of the artificial,* *including the study of self-reproduction in artificial systems.* Although these concepts are clearly fundamental to biological evolution, even within biology there is still ongoing controversy and current research into appropriate definitions for these concepts [43].

Coming back to Darwin's ideas, some degree of *heritability of fitness* is required for nontrivial evolution to occur. With self-reproduction, the similarity of offspring to the parents and the similarities of the environments in which the replicators find themselves is often enough to account for this. However, beyond the level of simple replicators, heritability of fitness requires more explanation, e.g. in considering multicellular lifeforms with differentiated cell types, subunits which are themselves replicators comprise populations within the body that are themselves potentially subject to evolutionary pressures [11, 45, 47, 48]. For example, cancer is an example in which reproduction and evolution occur at the lower cellular level at the expense of the higher organismal one. Multicellularity can arise (in certain conditions on mutations and cost of defection) where fitness (reproductive success) at the higher, whole organism level emerges in a trade-off against short-term fitness at the lower, cellular level. *Guaranteeing that the offspring are similar to the parent by suppression of freedom at the lower level in exchange for benefits is the first functionality required of any higher unit of fitness such as a multicellular organism.* The latter must employ mechanisms to balance the tendency of the lower level to defect by sufficient benefits from cooperation in the higher level unit, in order to persist over evolutionary time [48, 47].

*In asynchronous self-reproduction the very fact that the relative synchronization of the entire state of the "organism" is uncertain contributes to this problem of heritability of fitness.*

### 5.4 Self-Repair: Biological Methods and Generalizations

Self-reproduction and self-repair (or self-maintenance) are often closely related in biology, and an understanding of self-reproduction can thus contribute to our ability to create self-repairing, self-maintaining hardware and software. Von Neumann [98] considers synthesis of reliable organisms from unreliable components through redundancy and degeneracy, but apparently did not extend this during his lifetime to self-repair or relate it directly to self-reproduction. Automata models and circuitry capable of autonomous fault-detection and self-repair is an increasingly important area [40, 89, 41], both as an means to understand principles of biological organization, and also in technological applications, including robust computation, and especially for mission critical systems in space sciences. The capacity of a system to generate parts and components of its own structure and to establish their organization might obviously be useful in generating and installing a replacement parts in maintaining itself.

### 5.5 Self-Maintaining and Self-Creating Systems: Autopoiesis

Such a capacity for production of constituent components in the building and maintenance of them in an organized structure and dynamical process in the face of favorable or unfavorable perturbations (such as damage, production of waste, and entropic decay) is identified, according to biologists F. Varela and H. Maturana, as the key property, *autopoiesis* (“self-production”), defining living systems [95].

Neither von Neumann’s work on self-reproducing automata, nor the studies following him have addressed via constructive models this aspect of living systems. Langton’s work on self-reproducing loops (removing the universal construction and universal computation capacity) and its successors have focused on minimal models of self-reproduction, first by minimizing the size of replicators [12, 71, 37], and then adding various computational and other abilities [86].

**Autocatalysis, Compartmentation, Early Life** An *autocatalyst*, by definition, promotes its own formation from other materials, and thus is in some sense self-replicating [64]. Autocatalysis implies dynamical cycles, potentially continuing without end. Compartmentation proceeds via isolation of an environment inside a vesicle or membrane (see Figure 2) within which conditions are conducive to the autocatalytic cycle and the production and maintenance of the membrane. Self-replication in early life might thus have arisen as a bifurcation in the dynamics of a self-producing, self-maintaining system resulting in response to some perturbation. an early self-maintaining vesicle is broken into two parts along its membrane; each surviving component repairs itself comprising a new self-producing organization.[⋆] For example, due to increase in size or due to accidental damage, Any heritable aspect of organization that increases stability following such an event leads to similar descendants, potentially growing exponentially in number.

### 5.6 Challenge Problems

Work in constructive biology and the theory of self-reproducing automata discussed above leaves several challenges unanswered:

**Open Problem 2.** Realize construction universality in any computationally feasible, implementable models.

**Open Problem 3.** Construct an autopoietic self-reproducer whether synchronous or asynchronous in logical, kinematic or physical realization.

**Open Problem 4.** Solve open problem 3, adding heritable variation to realize evolution in a population of autopoietic self-reproducers.

**Genetic Acquisitions** Sex in biology is, by definition, nothing more than the transfer or exchange of genetic material. It occurs, e.g., between homologous chromosomes in meiosis, or in the uptake of DNA from the environment by bacteria. If precious genetic information is lost due to damage to DNA, or if an organism is doing poorly due to

[⋆] [cf. N. Ono and T. Ikegami, Artificial Chemistry: Computational Studies on the Emergence of Self-Reproducing Units. In *Advances in Artificial Life* (J. Kelemen and P. Sosı́k (Eds.), Springer Lecture Notes in Artificial Intelligence, vol. 2159, 2001 ) pp. 186–195].

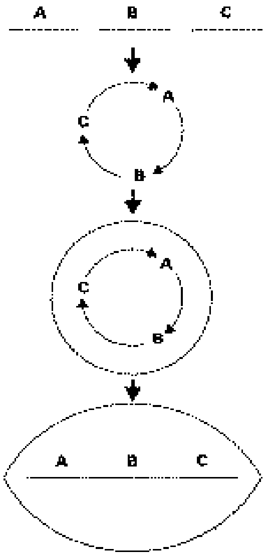


**Fig. 2.** Compartmentation as Proto-Self-Maintenance: Components of an Autocatalytic Cycle in a Protected Environment with only Constituent Partners present in suitable proportions. (After J. Maynard Smith [44])

heavy environmental stress, recourse to the genetic material from others may save the day by providing an undamaged source – though quite possibly different in content – of relevant genetic information; see [46] for the role of sex in repair.

A more extreme acquisition of genetic material than in sex is the acquisition of entire genomes in *symbiogenesis* (the advent of a merged entity, derived from replicators from two evolving populations, which becomes the unit of selection in an evolutionary process – see Appendix), and resulting speciation [43]. Such processes were involved in the acquisition by eukaryotes of the bacterial ancestors of mitochondria and, in plants, of chloroplasts [42].

**Open Problem 5.** Construct an evolutionary system in which different populations of autopoietic self-reproducers interact, and in which one species acquires the genome of another, realizing symbiogenesis.

### 5.7 Evolution of Evolvability

Finally, how can evolvability itself evolve? Lineage selection arguments suggest that lineages will survive that are robust to variational operators acting an evolving population [66, 5, 91]. The genetic code and genotype phenotype mapping, and genetic switches have all arisen in organic evolution of life on earth.

**Open Problem 6.** Construct an evolutionary system in which the capacity of populations to generate adaptive heritable genotypic and phenotypic variation increases without bound.

Evolution of developmental genetic regulatory networks in constructed artificial systems interacting with their environments (see sec. 8) is suggested as one road toward achieving some small reflection of what nature has achieved.

## 6 Local Time

We adopt here the view of local time as a random variable to approach the asynchronization problem for automata networks. That is, given a synchronously updating network of automata, we want to construct another network of automata, with essentially the same behaviour, but in which at each node logical time is determined by an (unknown) local random variable. It is at first unclear whether this is possible at all, since simple experiments with common cellular automata networks show that the behaviour of the system generally changes radically in a qualitative sense when abandoning synchronous update.

An **automata network** consists a collection of automata $\mathcal{A}_v$ associated to the vertices $v \in V$ of a locally finite directed graph $\Gamma = (V, E)$, and a global input alphabet $X$ and local transition rules $\delta^v$. A state of the network is a choice of state for each component automaton. Given a global input $x \in X$ and a state of the automata, the *next state of the network at node* $v$ is determined by the state of the automaton at $v$, the states of the automata in the neighborhood of $v$ (i.e. at those nodes $w$ which have an edge $(w, v) \in E$ to node $v$), and $x$. Thus the new state of the automaton at node $v$ may be written as

$$q^{v\prime} = \delta^v(q^v, q^{N(v)}, x),$$

where $q^v$ and $q^{N(v)}$ are, respectively, the current state at $v$ and the states $q^w$ of all nodes $w$ in the neighborhood of $v$.

An automata network is *synchronous* if every node advances to its next state simultaneously. Otherwise it is called *asynchronous*.[9]

[9] We assume for asynchronous update that there are no delays in state information reaching a node in a local transition and that local updates may be regarded as instantaneous. We do *not* require any particular ordering of updates of nodes, only that, after an update of any given node, each node will still be updated an unbounded number of times in its future. Simultaneity of the update of any two nodes is permitted (but not required), and massive asynchronous parallelism is thus possible.

We may assume an ambient physical time in which stochastic update events occur, i.e. particular subsets of the sets of nodes are updated at discrete moments of physical time; every node is updated an unbounded number of times; and no node is updated an infinite number of times within a bounded interval of physical time.

In our model of asynchronous networks, based solely on a function of its local neighborhood and state information, a local automaton may choose to read or delay reading the next letter in global input sequence. Reading of the global input sequence is thus not synchronized but happens independently at each node.

See [56] or [21, Ch. 7] for more details and proofs of theorems stated here.

An automata network is called a *cellular automaton* if it has only one global input letter (i.e. the alphabet satisfies $|X| = 1$ and its unique letter can be considered a “clock tick”), and the local transition functions, local automata, and neighborhoods at each node are isomorphic. Synchronous cellular automata have been well-studied since they were introduced by S. Ulam and J. von Neumann in the middle of the last century (e.g. [99, 13, 10, 88])

**Definition (Emulation).** Let $\mathcal{A}$ be an synchronous automata network over a directed graph $\Gamma = (V, E)$ with global state set $Q$ and $\widehat{\mathcal{A}}$ be an asynchronous automata network with the same input alphabet $X$, a directed graph $\Gamma' = (V, E')$ with the same set of nodes, and global state set $\widehat{Q}$. Let $\pi : \widehat{Q} \to Q$ be a function from global states of the asynchronous automata network to global states of the synchronous one, such that $\pi^v(\hat{q}) = (\pi(\hat{q}))^v$ depends only on $\hat{q}^v$ for all $\hat{q} \in \widehat{Q}$. Thus we can denote $(\pi(\hat{q}))^v$ by $\pi(\hat{q}^v)$.

Regarding physical time as modeled by non-negative real numbers and logical time in the synchronous automata network as modeled by the natural numbers, we then say that the behavior $\hat{q} : \mathbb{R}^+ \to \widehat{Q}$ of $\widehat{\mathcal{A}}$ starting in state $\hat{q}_0$ for update pattern determined by local random variables at each node (as above) and input sequence $x_1, x_2, \ldots$ ($x_i \in X$ for $i \in \mathbb{N}$) *emulates* the behavior $q : \mathbb{N} \to Q$ of $\mathcal{A}$ starting in state $q_0$ with the same input sequence under the projection $\pi$ if there exists a *spatial-temporal covering* $\lambda : \mathbb{R}^+ \times V \to \mathbb{N}$, i.e. the following diagram commutes for each $v \in V$:

$$\begin{tikzcd}
\mathbb{R}^+ \arrow[r, "\hat{q}^v"] \arrow[d, "\lambda(-,v)"'] & \widehat{Q}^v \arrow[d, "\pi"] & \text{(asynchronous)} \\
\mathbb{N} \arrow[r, "q^v"] & Q^v & \text{(synchronous)}
\end{tikzcd}$$

That is, $\pi(\hat{q}^v_t) = q^v_{\lambda(t,v)}$, with $q^v_n =$ state in $\mathcal{A}$ of node $v$ at time $n \in \mathbb{N}$ and $\hat{q}^v_t =$ state in $\widehat{\mathcal{A}}$ of node $v$ at time $t \in \mathbb{R}^+$.

Thus the behaviour of $\widehat{\mathcal{A}}$ projects onto and completely determines the behaviour of $\mathcal{A}$.

**Theorem 1 (Emulation by Asynchronous Automata Networks [56]).** *Let any synchronous automata network $\mathcal{A}$ over a locally finite digraph $\Gamma = (V, E)$ with local automata $\mathcal{A}^v = (Q^v, X^v, \delta^v)$ ($v \in V$) and external input alphabet $X$ be given.*

*We construct an asynchronous automata network $\widehat{\mathcal{A}}$ (with the same input alphabet $X$) such that every possible behavior of $\widehat{\mathcal{A}}$ with input sequence $\{x_n\}_{n>0}$ emulates the (only possible) behavior of $\mathcal{A}$ with input sequence $\{x_n\}_{n>0}$, when $\widehat{\mathcal{A}}$ starts in an initial global state $\hat{q}_0$ depending only on the initial global state $q_0$ of $\mathcal{A}$.*

*Moreover, the following hold:*

1. *The underlying digraph for $\widehat{\mathcal{A}}$ is the reflexive-symmetric closure of the digraph for $\mathcal{A}$.*
2. *For each vertex $v$, the local automaton $\widehat{\mathcal{A}}^v$ at vertex $v$ in $\widehat{\mathcal{A}}$ is “not much more complicated” than the local automaton $\mathcal{A}^v$ at $v$ in $\mathcal{A}$. Indeed, $\widehat{\mathcal{A}}^v$ is a product of $\mathcal{A}^v$, an identity-reset automaton, and a modulo three counter. In fact, $\mathcal{A}^v$ has state set $\widehat{Q}^v = Q^v \times Q^v \times \{0, 1, 2\}$.*

3. *The projection* $\pi : \widehat{Q} \to Q$ *is given locally by* $\pi^v(q^v, b^v, r) = q^v$ *for* $(q^v, b^v, r) \in \widehat{Q}^v$*.*
4. *The starting state of* $\widehat{\mathcal{A}}$ *is given by* $\hat{q}_0^v = (q_0^v, q_0^v, 0)$ *for all* $v \in V$*.*
5. *Furthermore, the spatial-temporal covering of the emulation satisfies*

$$|\lambda(t, v) - \lambda(t, v')| \leq \lfloor \frac{d(v, v') + 2}{3} \rfloor,$$

*where* $d$ *is the distance metric in the graph* $\widehat{\Gamma}$*.*

Note that updates of local states in the constructed emulating automaton are essentially arbitrary.

We call $\lambda(t, v)$ the *local time* of the synchronous automaton $\mathcal{A}$ at vertex $v$ for time $t$ in the emulating asynchronous automata network $\widehat{\mathcal{A}}$. Of course, $\lambda$ depends in general on the update pattern for the particular behavior of $\widehat{\mathcal{A}}$. Thus (5.) above says that the difference in local time at two nodes in the emulating asynchronous automata network is bounded above by approximately one third of the distance between them.

*Brief Sketch of Proof and Construction:*

Let $N(v)$ denote the set of neighbors of node $v$, and let $\widehat{N}(v)$ denote the neighbors of $v$ in the reflexive-symmetric closure of $\Gamma$, which gives the topology of the emulating asynchronous automaton $\widehat{\mathcal{A}}$. The local update function in $\widehat{\mathcal{A}}$ is defined as follows, where $\varphi^v$ and $\widehat{\varphi^v}$ give the action at vertex $v$ in $\mathcal{A}$ and $\widehat{\mathcal{A}}$, respectively, as a function of their arguments, depending only on local state in the neighborhood and global input letter:

$$\widehat{\delta}^v((q^v, b^v, r^v), \widehat{\varphi}^v(\hat{q}, x)) =$$

$$\begin{cases} (q^v, b^v, r^v) & \text{if } r^w = r^v - 1 \bmod 3 \text{ for some } w \in \widehat{N}(v) \\ (q^v, b^v, r^v + 1 \bmod 3) & \text{if } r^w \neq r^v - 1 \bmod 3 \text{ for all } w \in \widehat{N}(v) \\ & \quad \text{and } r^v \neq 0 \\ (q^v \cdot \varphi^v(c, x), q^v, 1) & \text{otherwise,} \end{cases}$$

where $c$ be an arbitrary state of $\mathcal{A}$ such that for each $w \in N(v)$,

$$c^w = \begin{cases} q^w \text{ if } r^w = 0 \\ b^w \text{ if } r^w = 1. \end{cases}$$

Note each $r^w$ must lie in $\{0, 1\}$ in determining $c^w$ of the third case, as necessarily $r^v = 0$ in third case and $w \in N(v) \subseteq \widehat{N}(v)$ implies $r^w \neq 2 \mod 3$.

Thus, in the emulating automata network the neighboring nodes carry a copy of “current state” and “old state” in case a neighbor needs to read either one. The third component of state carries a modulo 3 value. The neighbors of any node $v$ can be shown inductively to receive the same number of increments modulo 3 as node $v$, plus or minus one. Thus neighboring nodes differ by at most 1 modulo 3 in this component. In computing its local update, a node can check whether each of its neighbors is in the past, future, or in sync with it. If any neighbor is in the past, no update is performed (and the global input letter is not read). Otherwise, we increment the modulo 3 counter and on every third counter increment, copy current state to old, and update the current state according to the update rule of $\mathcal{A}$ and the global input letter. In the latter case, every neighbor must be in sync or in the future relative to the node in question, so the appropriate state of the neighbor node in $\mathcal{A}$ can be determined from the current or past state component of the corresponding neighbor in $\widehat{\mathcal{A}}$.

Using the fact that nodes differ by at most one in the number of increments they receive in the third component and using local finiteness another lemma shows *freedom from deadlocks* – a node can only be waiting for one that has received one less such increment and only finitely many can have occurred, so any chain of waiting ends when the automata at its end (with fewest increments so far) receives an update. Induction then shows that behavior of the synchronous automata network can be recovered uniquely from any behaviour of the asynchronous one by a *spatial-temporal section* $\lambda(t, v)$ equal to the ceiling of the one-third of two plus the number of counter increments at node $v$. (See [56] for full details.) □

A special case of essentially this construction was found independently and presented by K. Nakamura [51], the author [55, 53], and T. Toffoli [87, 88], with full rigorous proof of its correctness given in [56]:

**Corollary 1 (Asynchronous Emulation of Cellular Automata Networks Theorem).** *If $\mathcal{A}$ is a synchronous cellular automaton then there is an emulating asynchronous cellular automaton $\widehat{\mathcal{A}}$.* □

**Open Problem 7.** Prove an analogue of the Asynchronous Emulation Theorem for Automata Networks that may dynamically change their topology and number of component automata. (Or, more weakly, prove such an analogue for cellular automata networks.)

### 6.1 Temporal Waves, Asynchronous Game of Life and Universal Computation

**Temporal Waves** From what we saw in the last section, it follows that local time in the asynchronous emulating network for nodes at distance $d$ differs by at most about a third of the distance between them.[10] Since the values of the modulo 3 synchronization counter differs by at most 1 between neighbors in the asynchronous emulating network, this spatial continuity of the modulo 3 counter state entails that updates corresponding to simultaneous ones in the synchronous network move as temporal waves across the space of the asynchronous network.

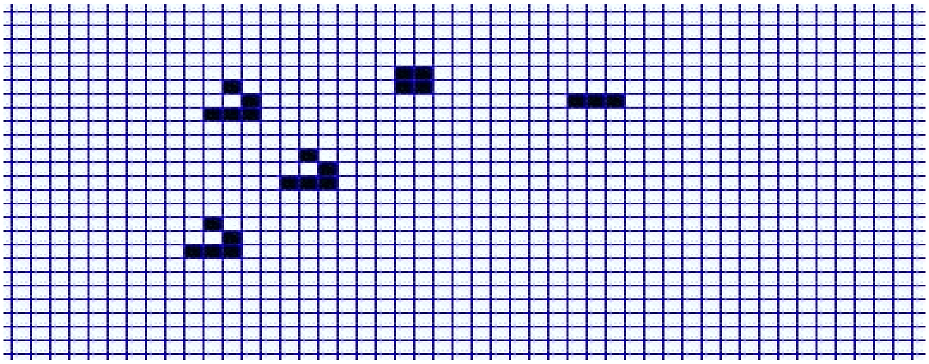
**Asynchronous Game of Life** This phenomenon is illustrated here with an asynchronous version of John Conway's famous synchronous cellular automata network, "The Game of Life".

Let us apply the construction to Conway's (synchronous) Game of Life. A local automaton in synchronous Life has two possible states (quiescent (0) or alive (1)) and the

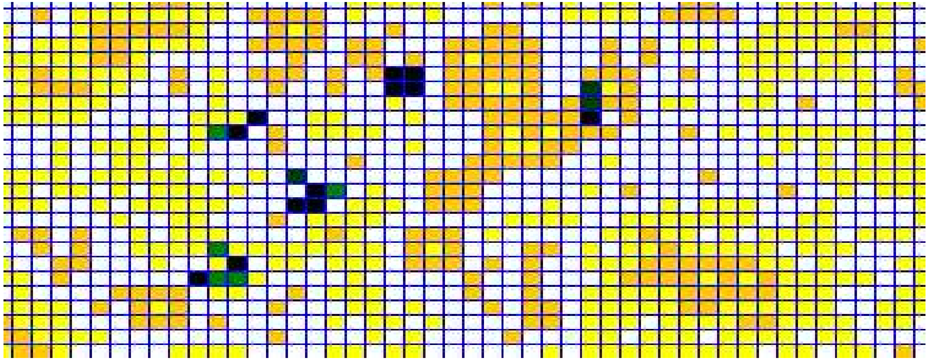
[10] Disconnected components are of course at infinite distance, and so the temporal disparity between them can be arbitrarily large.

ASYNCHRONOUS GAME OF LIFE

Initial State:



Progress of Gliders in Asynchronous Life. Note that the upper left hand glider is not recognizable as one due to small local temporal variation in its cells:



Further Progress of Gliders in Asynchronous Life. All their parts are nearly in the same spatial-temporal section; all three gliders are now recognizable again:

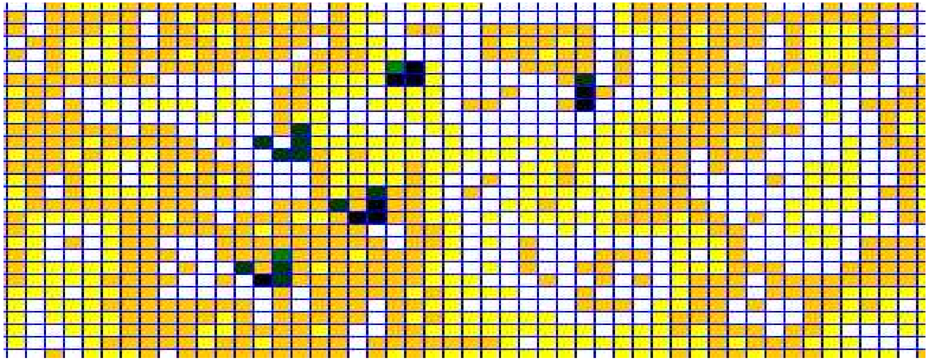


**Fig. 3.** Temporal Waves and Progression of 3 Gliders, with Box, and Blinker in Asynchronous Game of Life. Contiguous regions of the same shade are "temporal wavefronts" that represent the same moment in a spatio-temporal section giving the global state of the corresponding synchronous cellular automaton [55]. Shade is determined by value of modulo 3 counter at a given node. Neighbor nodes differ by at most one time unit with respect corresponding nodes in the synchronous model.

following transition function: if a cell is quiescent and has exactly 3 neighbors that are alive, its next state is alive. If a cell is alive, and it has either 2 or 3 live neighbors (not including itself) then it stays alive, otherwise it becomes quiescent. It is well-known that, in principle, universal computation can be implemented in a infinite two-dimensional (synchronous) cellular automaton running Conway's rule (for an enjoyable yet detailed overview see chapter 1 of [81]).

Figure 3 (top panel) shows an initial configuration of some well-known structures in Conway's Game of Life as an initial configuration for the corresponding asynchronous cellular automaton: Three gliders which move across the space, a stable $2 \times 2$ box, and a blinker (a row of 3 cells, that becomes a column of 3 cells, then a row of 3 cells, and so on).

The next panel shows the state of the world a few time steps later, the shading indicates the synchronization state of the cell in the space, while the darker cells of various shades are live cells in various stages of temporal synchronization. Contiguous cells of the same shade are in sync and reflect the same instant of time in the synchronous cellular automaton. The third panel down shows the state of the system a little later.

**Asynchronous Universal Computation** The possibility of implementation of Conway's Game of Life in an asynchronous cellular automaton as illustrated here entails that universal computation is possible in a two-dimensional asynchronous cellular automata running the modified rules (see [81] for a lively exposition).

Of course, a Turing machine can be regarded as a synchronous 1-dimensional cellular automaton where all state transitions are trivial except in the vicinity of the read-write head. Thus applying Corollary 1 to a universal Turing machine also yields the result.

### 6.2 Asynchronous Self-Replicators

Applying the construction of the theorem to Langton's self-reproducing loop, and numerous self-reproducers including those of Byl, Reggia et al., Sayama and others mentioned above, we implemented the first asynchronous self-reproduction in cellular automata [55]. Figure 4 illustrates asynchronous replication of a structurally dissolvable loop capable of programmed cell death.

## 7 Minimal Evolvable Systems

To better understand evolvability we considered some open-ended evolutionary systems. Now we examine two (more or less) minimal evolvable systems to study how evolvability, and in particular variability, can arise.

### 7.1 Minimal Example 1: Asynchronous Evoloop

This example is due to Sayama-Nehaniv [77, 53] by combining their techniques. Here a population of self-replicating loops in finite space is implemented (asynchronous cellular automata; physics: changes according to deterministic rules depending on local neighborhoods; asynchronous version of Sayama-Langton evoloop [53]).

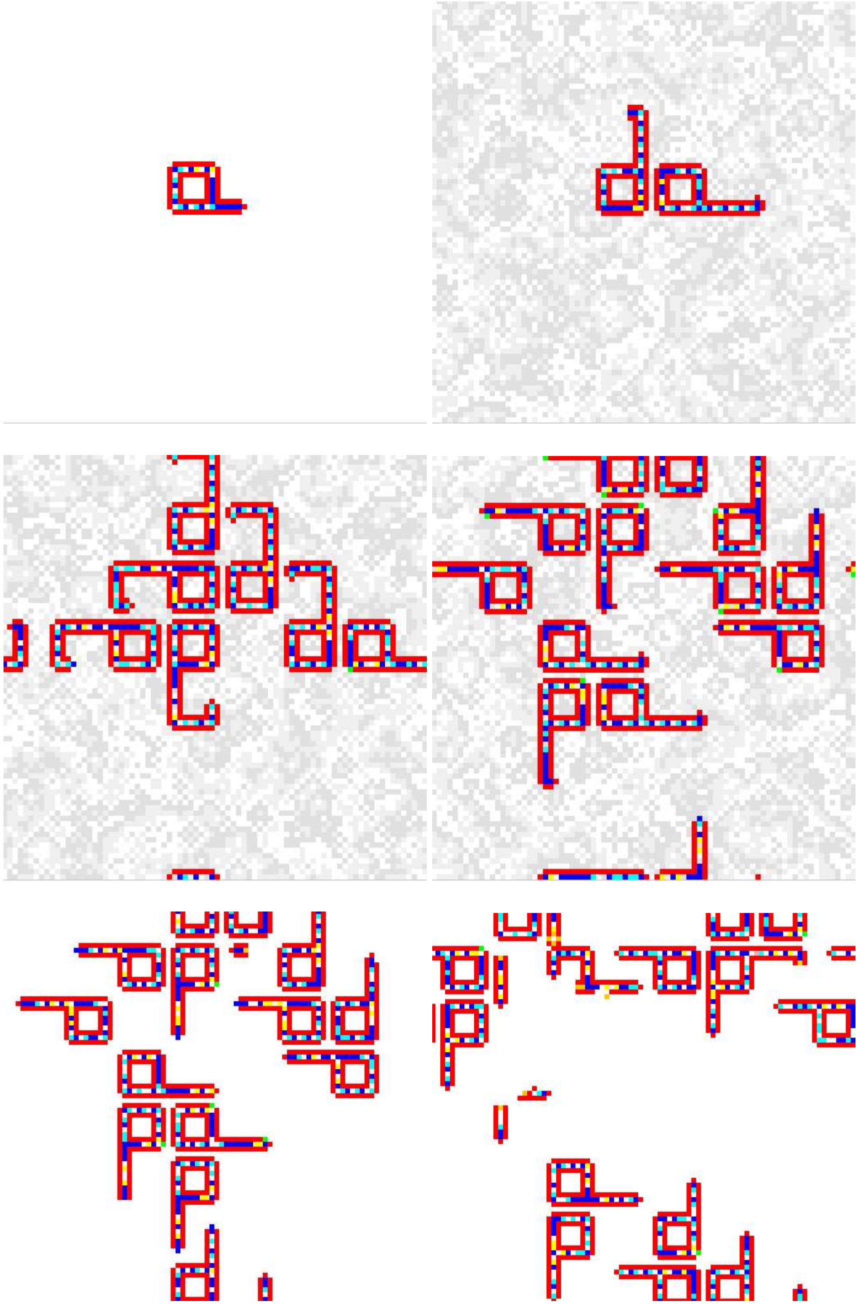


**Fig. 4.** Asynchronous Version of Sayama's Structurally Dissolvable Self-Reproducing Loop. Space is liberated by "programmed cell death" and can be reused by descendents of the original loop (6 snapshots of a single run; toroidal topology). Differences in shading (shown only in the first four panels) correspond to differences in the synchronization component of local state (cf. discussion of temporal waves).

Sayama [76], extending Langton's construction, introduced apoptosis. Apoptosis ("programmed cell death"), locally started, is triggered by local rules in response to stagnant or unexpected configurations (tending to indicate non-viability) generating a suicide signal, which propagates over to contiguous local automata that are non-quiescent [76]. This results in resource freeing and makes possible the turn over of generations required by evolution.

A further synchronous variant, evoloop, allows evolution in a cellular automata network to be realized [77]. By careful design of the update rules, ancestral self-reproducing loops are robust to some interactions (collisions) with others in space. They might either recover from a collision with another loop, undergo an apoptosis chain reaction, or survive in a changed form. The latter may or may not have the same circulating genome determining the construction of its potential offspring. If a changed loop produces viable, reproductive offspring, then variation is inherited, so variability has been introduced in evolution.

Applying the asynchronous emulation theorem yields the first implemented example of an asynchronous cellular automata network with the capacity for Darwinian evolution (minimal evolvability), including heredity, variability, differential reproductive success, finite resources and turn-over of generations [53].

Over evolutionary time, loops of different sizes arise; smaller loops can replicate more quickly and are less likely to collide than large ones; the population generally evolves smaller and smaller loops until no further reduction in size is possible. See Figure 5.

**Sources of Variability: Interaction.** Interactions during lifetime are the major selective force but also the source of variation. There is only limited potential for variability (rotation of genetic core; loop and genome size).

### 7.2 Minimal Example 2: Cultural Evolution in Alissandrakis' Imitating Robotic Arms

Another instance of evolution occurs in human and non-human culture with the transmission of patterns of behavior (or *memes* [19]). Imitation broadly construed is the transmission mechanism for memes.[11]

**Imitation, Social Learning, and Cultural Evolution** Learning behaviours from others, with cultural variations between populations that are not explainable simply due to

[11] There is a unsettled debate on whether a meme should be regarded as an unobservable (at least until now) pattern of information in the brain or as an observable expressed pattern of behaviour. The former seems more "genotypic" (as an informational pattern, but it probably is incapable of ever being directly copied from one individual to another) and the latter more "phenotypic" (as an effect of such a pattern).

Moreover, it is unclear what constitute an *individual* meme in the population dynamics of memetic evolution, or when two memes are "the same" (either at the neuronal or behavioral level).

Evolvability (First Asynchronous Cellular Automata Network Example)

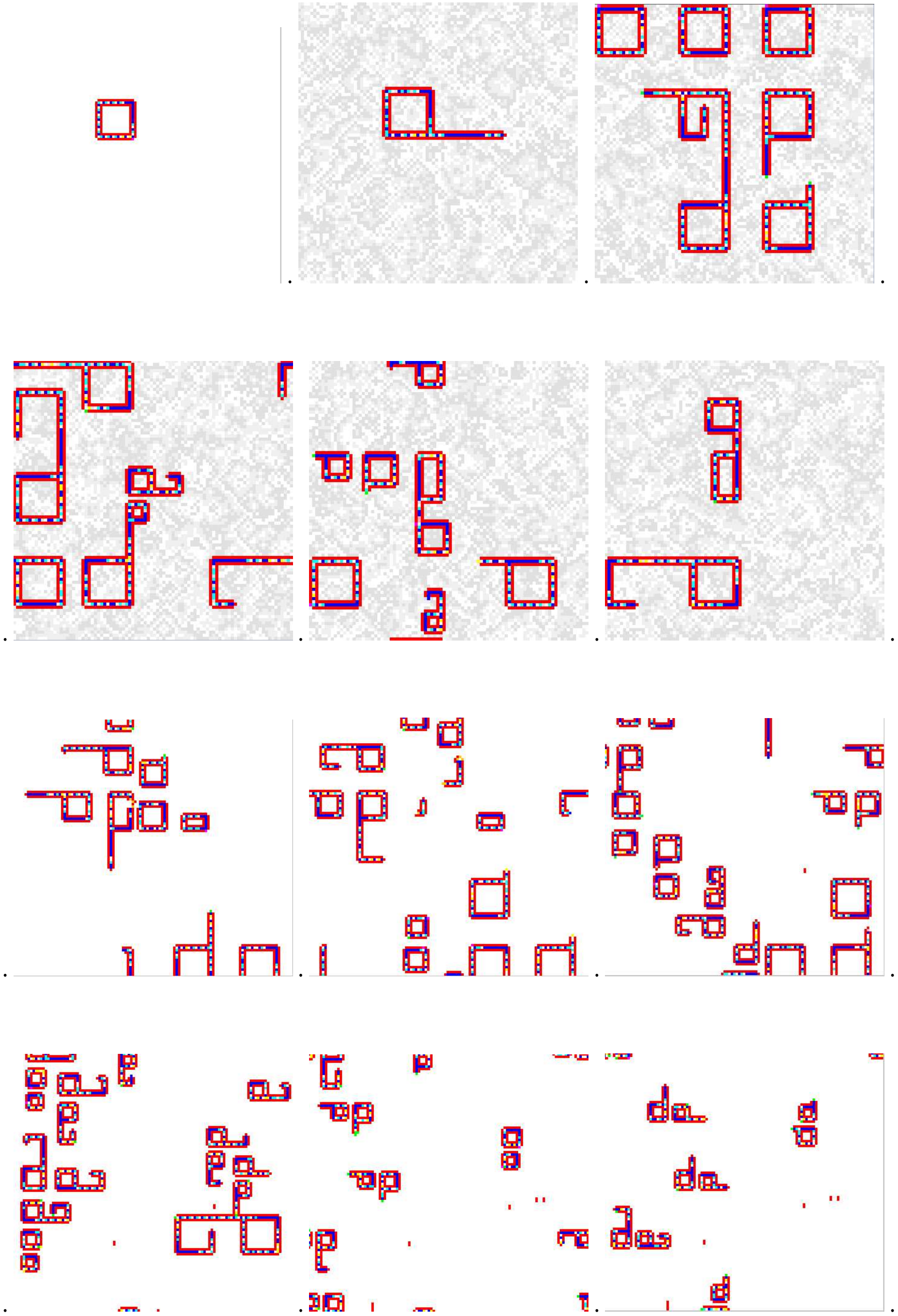


**Fig. 5.** Evolution in Asynchronous Cellular Automata: Asynchronous Version of Self-Reproducing Evoloop. (12 snapshots (not all from the same run); toroidal topologies). Heritable variability of characteristics of individuals (e.g. loop size) entails that this is an evolutionary system. Evolution leads to small, fast-replicating loops that are less likely to collide than larger ones. Temporal waves shading is shown in the first six snapshots [53].

differences in local ecological context, has been established not only for humans, but also in some other animal species, including cetaceans [72] and in chimpanzees [105].

Cultural evolution is based on transmitted patterns of behavior, and is exhibited by humans and some other animal species. Social learning, imitation, and/or instruction allow an organism to learn from the experience of others, which facilitates the accumulation of cultural practice and obviate much, often dangerous, trial-and-error individual learning. Social learning can also be combined with individual learning to exploit creative variability. In several realms (behaviors, technology artifacts, language) cultural evolution can be open-ended.

Cultural (memetic) evolution is possible in artificial societies and in the future might find application e.g. in factories populated by autonomous robots of various types who acquire and transmit skills and task knowledge through social learning. These robots could acquire skills and competencies by observing others (e.g. human demonstrators, or industrial robotic arms with different sizes, kinds and numbers of joints) and pass them to newcomer robots of as yet unknown type when they join the population.

A simple robotic population model illustrates this potentiality [3, 4]: Simulated robotic arms are used, with differing lengths of segments, differing numbers of joints, but all with fixed base about which they can rotate (Figures 6 & 7). The robot arms carry out behaviors in a two-dimensional workspace and engage in social learning via imitation. The agent embodiment can be described by a vector $L = [\ell_1, \ell_2, \ell_3 \cdots, \ell_n]$, where $\ell_i$ is the length of the $i^{\text{th}}$ joint. Each robot builds a "correspondence library" to

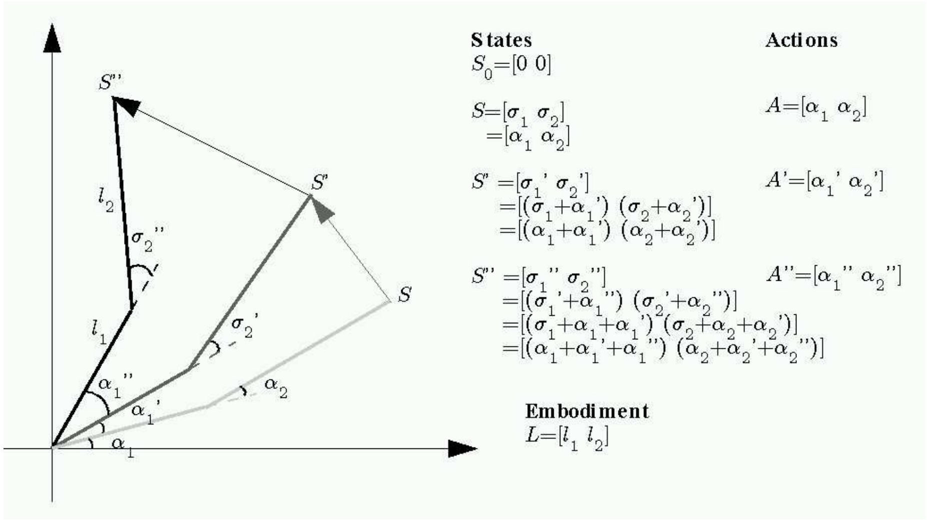

$S''$
$S'$
$S$
$l_2$
$\sigma_2''$
$l_1$
$\sigma_2'$
$\alpha_1''$
$\alpha_1'$
$\alpha_2$
$\alpha_1$
States
$S_0=[0\ 0]$
$S=[\sigma_1\ \sigma_2]$
$=[\alpha_1\ \alpha_2]$
$S'=[\sigma_1'\ \sigma_2']$
$=[(\sigma_1+\alpha_1')\ (\sigma_2+\alpha_2')]$
$=[(\alpha_1+\alpha_1')\ (\alpha_2+\alpha_2')]$
$S''=[\sigma_1''\ \sigma_2'']$
$=[(\sigma_1'+\alpha_1'')\ (\sigma_2'+\alpha_2'')]$
$=[(\sigma_1+\alpha_1+\alpha_1')\ (\sigma_2+\alpha_2+\alpha_2')]$
$=[(\alpha_1+\alpha_1'+\alpha_1'')\ (\alpha_2+\alpha_2'+\alpha_2'')]$
Actions
$A=[\alpha_1\ \alpha_2]$
$A'=[\alpha_1'\ \alpha_2']$
$A''=[\alpha_1''\ \alpha_2'']$
Embodiment
$L=[l_1\ l_2]$

**Fig. 6.** An example robotic arm imitating agent. A two-joint robotic arm, with segments of length $\ell_1$ and $\ell_2$, moving from *state* $S_0$ (arm completely outstretched along the horizontal axis) to state $S$ to state $S'$ to state $S''$, as it sequentially performs *actions* $A$, $A'$, and $A''$. Effects of these actions (marked trails) are shown as the arrows that join the tips of the arm as it moves.

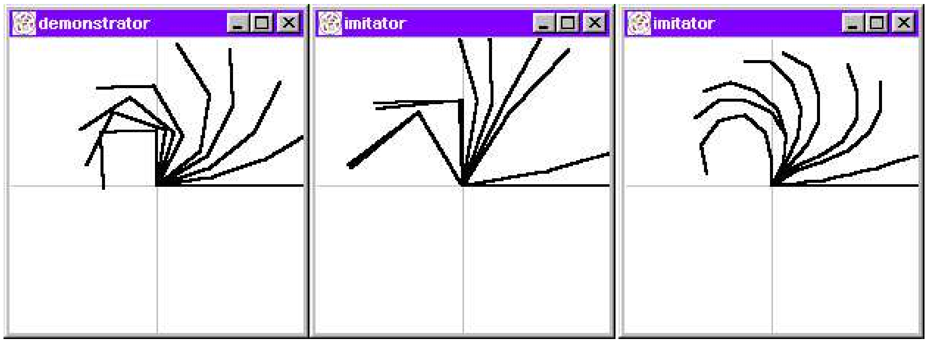


**Fig. 7.** Solving a correspondence problem for matched behaviour between different embodiments. A demonstrator behaviour consists of a model folding its 3 joints counter-clockwise (left). Imitation attempts to match the position of the end point are shown for a 2-joint imitator (center) and a 6-joint imitator (right).

imitate another, possibly dissimilar one (using various metrics of similarity to reinforce success). A robot arm observes another one (with possibly different embodiment) and attempts to match its behaviour (according to some metric such as posture, end-effector position, or angle changes at the joints). In turn, a third robot arm observes the imitator and attempts to imitate it (again using its own possibly different embodiment, using some metric), but does not observe the first robot, and so on. Behaviors can thus be culturally transmitted through a chain of robots.

The example illustrated in Fig. 8 demonstrates such (horizontal) transmission of a behavioral pattern via social learning in a chain of imitating agents. The original model with three joints is shown in (Fig. 8, left). It is imitated by a two-joint robotic arm (Fig. 8, center), which in turn is imitated by another imitator (Fig. 8, right) with the same embodiment as the original model, but which only perceives the behavior of the two-joint agent. After transmission through the intermediary, the behavioral pattern that has been acquired by the second imitator in (Fig. 8, right) is quite similar to the original despite differences in embodiment in the chain of transmission. This example illustrates transmission of a behavioral pattern through a chain of robotic agents, despite differences in embodiment of agents involved. This simple example serves as proof of the concept that by using social learning and imitation, rudimentary cultural transmission with variability is possible among robots, even heterogeneous ones.

Evolutionary emergence of shared behavior and rudimentary 'proto-culture' in populations of robotic arms is discussed in [4]. Figure 9 shows imitators, with different embodiments arranged in a circle each learning by imitating its neighbor, and the resulting emergence of shared behavior.

Synchronization (via resetting to a fixed initial posture) before each demonstration has been shown to generally result in much faster and more accurate behavioral transmission [3].

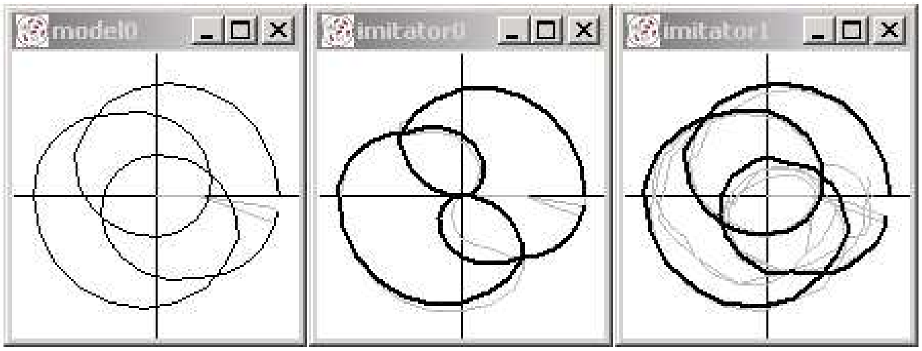


**Fig. 8. An example of social transmission.** Trail left by end-effector tips during behaviors by 3 robot arms are visualized. The original model `model0` ($L = [20, 20, 20]$) is shown to the left. In the middle, a two-joint `imitator0` ($L = [30, 30]$) acts also as a model for `imitator1` on the right ($L = [20, 20, 20]$)). Due to the different embodiment of the agent `imitator0`, the replication of the model pattern is similar, but not exact. `imitator1` has the same embodiment as the original model `model0` and, although indirectly transmitted, the resulting pattern is closer to that of the original model than is the behavior of the intermediate agent `imitator0` used as a model by this second imitator [3].

**Sources of Variability: Embodiment Differences.** The behaviors of the arms are the selectable entity for a Darwinian evolutionary process: Imitation is the replication mechanism for these behaviors. Resources are finite since there are finitely many arms and each arm can only perform one behavior at a time.

Variability arises from several sources: (1) Errors in observation and noise in production of a behavior can introduce variability in a behaviour, which an observing robot might match, learn and pass on. (2) Embodiment differences may constrain what an imitator can do. For instance, a complex folding up by a six-segment arm could not be matched exactly angle-for-angle by a three-segment arm. Conversely, the six-segment arm imitating a two-segment arm might vary the position of various joints in many ways and still achieve a satisfactory imitative behavior; further down the chain of transmission an observer of this six-segment robot arm might acquire some aspects of its behavior not present in the original behavior of the two-segment robot.

The first source of variability is very closely analogous to mutation at the level of copying errors and is not particularly novel. The second source of variability, *differences in embodiment*, is unlike what we know from biological or evolutionary computation examples.

Replication in the robot arm example is based on *interaction* (like in prions – proteins that can inherit a conformation from interaction with a variant protein – but with vastly more variability).[12]

[12] This is in contrast to evolving population of self-reproducing loops in example 1. There interaction provided the basis, not for replication, but for variability.

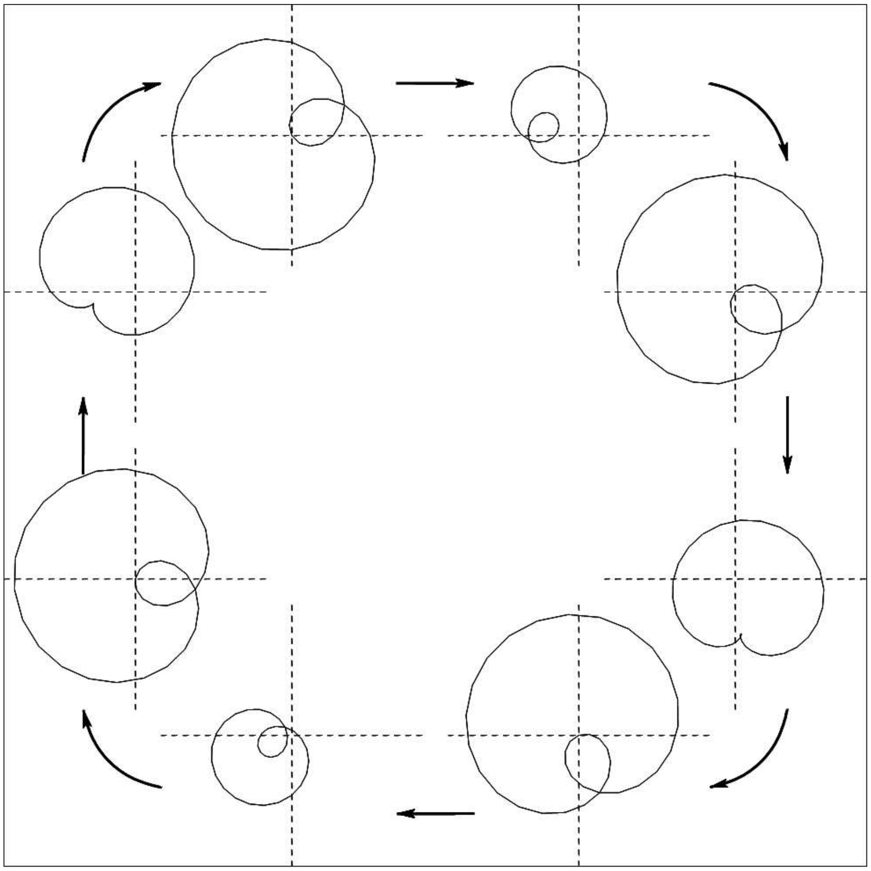


**Fig. 9.** Emergence of proto-culture among eight heterogeneous agents. Arm robots have alternating 2 and 4 joint embodiments (overall arm length remains constant). Starting with no initial “seed” model, and imitating each other clock-wise using a metric on actions, the figure shows an example of two stable repeated single-action variant behaviours emerging in the population: the agents with two joints move both joints anti-clockwise (by 10 degrees), while the agents with four joints freeze their first and the third joint, moving only the second and third joint. The different effect patterns shown result from the different states the agents are in when stable imitative behaviour is established.

## 8 Developmental Genetic Regulatory Networks (DGRNs)

A particular paradigm from nature realizes all of the above properties discussed for differentiation, duplication and divergence: developmental genetic regulatory networks (DGRNs). The most complex systems known to humankind are differentiated multicellular organisms. They may consist of e.g. on the order of between $10^{13}$ and $10^{14}$ cells in the human body; in multicellular organisms genetically identical cells differentiate, depending on the species, into between two or tens, or hundreds, of cell types [9] with each cell each capable of taking astronomically many states. The component cells themselves are each living entities each in itself already more complex than anything ever constructed by a human being. The organism maintains coherence as an individual while growing from a single cell into this huge number through asynchronous divisions, with a dynamically changing topology of interactivity between this changing number of cells including long, medium and short range interactions regulating patterning, global metabolism and the essential processes of life (see e.g. [106, 6]).

In nature genetic regulatory networks with development/differentiation are constantly engaged in interaction (within each cell, with the local environment of the cell, or at organismal level with the external, ecological, and possibly social environment). This kind of incessant activity in the control systems within each cell and their coherent integrated activity has been called *universal responsiveness* by West-Eberhard and lies at the basis of phenotypic and developmental plasticity [104]. Selection for robustness in development (due, for example, to pressures for fitness to be heritable from parent to offspring) could have as a non-selected by-product the following properties which enhance evolvability:
(1) phenotypic variation becomes tolerated and possible;
(2) particular phenotypic variations becomes heritable, since similar genes in a similar environment yield similar development; later canalized, and
(3) developmental versatility leads to increased phenotypic variability (along the "'right" dimensions of variation) serving as fodder for the next "rounds" of evolution.

Very little in Artificial Life has been achieved in the two modes of life involving self-reproducing autopoietic entities or symbiogenesis (see Appendix), but they can be approached on the foundations of current work. Genetic regulatory networks (GRNs) in cells are an essential component of how nature is able to grow developing, living systems [18]. GRNs are universally responsive control systems within biological cells. In multicellular organisms, GRNs are duplicated and diverge in functionality as organisms grow, in response to local conditions, the environment, and via signaling. They appear to provide essential properties for evolvability [1], and the continual, universal responsiveness and plasticity of living systems [104].

Operating continually in close connection with their environment through signaling channels, while actively maintaining internal dynamics, artificial GRNs easily allow for heritable digital genetic encoding, and provide a model analogous to that of a single cell (although presently without its replication capabilities). Unlike most other present-day computational models it is natural to apply them in an continually active and responsive mode [67]. Moreover, they exhibit very flexible evolvable, expressive dynamics similar to key biological regulatory phenomena useful in achieving a variety of control and computational dynamics [7, 67]. The evolvability properties of GRNs and DGRNs are being analyzed mathematically as dynamical systems using techniques, e.g. of [62, 61, 60], in efforts to develop a predictive theory of their evolution and application in novel computation, as well as Artificial Life. The genetic regulatory network in a developing organism is duplicated in each cell, which carries its own differing state (in cytoplasm, structural and epigenetic marking). Each cell has the same genetic network and responds to local conditions. Multicellular living organisms use DGRNs to control for growth and differentiation, as well as for incessant active control while growing from a single zygote (or "seed") to adult by cell division. The desirable dynamical systems properties of GRNs might be combined with development to allow flexible, responsive control, continually coupled to the environment in organisms consisting of even astronomical numbers of different cells. Massively parallel distributed, adaptive, robust, fault-tolerant, self-repairing control and computation is a hallmark of DGRNs in living organisms but very unlike what we find in conventional software engineering and von Neumann computation, but the potential of DGRNs for novel computation and the simulation and synthesis of life is only now beginning to be explored.

Reaction-diffusion, cellular signaling and positional information could be set up using tools available and being developed in evolving multicellular systems as a natural method for computational morphogenesis and novel, developmental computation.

Development of NETBUILDER, a test-bed for modeling the dynamics of multicellular genetic regulatory networks for biologists [78], is currently supported by a grant of the Wellcome Trust to Maria Schilstra and the author. It turns out that this platform can also be used to model artificial developmental genetic regulatory networks. Schilstra and Nehaniv [79] discuss the computational modelling of gene regulation in genetic regulatory networks, and current work is exploring the use of such computational networks to reverse engineer genetic regulatory control given gene expression data.

The evolutionary approach to understanding DGRNs is the most natural and would help characterize and evaluate aspects of their evolvability properties and developmental plasticity in different contexts.

Study of artificial versions of developmental genetic regulatory networks, comprising multicellular individuals, in evolving populations of such multicellular individuals is also a natural approach to addressing the essential questions of defining possible modes of life (see Appendix). Such evolving populations of DGRNs embodied in different environments may be rich enough to study (1) heritability of characters at higher levels, and (2) regulation of conflicts with the constituent cellular level (guided by some predictive theory from [47]), (3) emergence of self-maintenance at various levels, as well as (4) differentiation and modification of regulatory dynamics, and genetic, developmental and phenotypic plasticity.

Coupled with replication capability, evolving these artificial DGRNs (in software or in artificial proto-cells) could lead to systems showing more or all of the properties of life in its various modes. It would also be interesting to study (5) the induction of symbiogenesis in such systems, perhaps leading to artificial organelles and the degeneration of properties of life if capacity for independent maintenance of pattern integrity and replication is lost (mode 3). The genetic network in each cell of a differentiated multicellular organism is duplicated and diverges from its progenitors. Evolved differentiated multicellular organism possess a dynamic topology of interacting, developing genetic regulatory networks within their cells. These DGRNs have the following properties that could also be realized in implementations of artificial versions:

- Multiple copies of the same regulatory mechanism in similar units, with lineage structure
- Expressive and robust dynamical systems with parameters tunable by transcription factor (TF) binding strengths, concentrations, co-factors, etc.
- Layering of combinatorial logic on activation/inhibition of transcription ("biologic")
- Duplication-divergence via sensitivity of dynamics to epigenetic marking, development, environment, timing, cell-type, external signals

There are some open questions for GRNs and DGRNs:
(1) What is the degree of smoothness of their evolutionary dynamics?
(2) What is the relative importance of variability operators yielding regulatory changes vs. operators yielding gene product changes?
(3) What organizations of development yield what evolvability properties?
(4) What is the role of development and ecology in their evolvability (evo-devo-eco)?
(5) As biological DGRNs are naturally asynchronous, with no global clock coordinating their action, developmental and organismal time and timing must rely on local mechanisms to achieve coordination. The dynamic topology and asynchronous nature of DGRNs thus make them a promising test-bed for studying the evolutionary dynamics and emergence of asynchronous temporal coordination.

## 9 Conclusion and Major Challenges

Evolution has been presented as a powerful and general class of stochastic algorithms. Response to interactivity (phenotypic plasticity) with environment/others may be fundamental to evolvability. Interaction can play a selective and/or reproductive role in the capacity to evolve (as shown in two minimal examples exhibiting evolvability). Interaction can modulate duplication-divergence: Genetic regulatory networks, lifelong engagement, and differentiation/development appear to have important evolvability properties and consequences that deserve to be better studied. Interaction also plays a important role in the arising of multicellularity. Culture arises via social transmission of behavior, knowledge and skills, and is possible for constructed agents (e.g. robots on shop floor). In different example minimal evolutionary systems, interaction and embodiment can serve as a sources of variability. Developmental Genetic Regulatory Networks (DGRNs) are proposed as a paradigm for novel computation and the study of evolvability. Evolvability of autopoietic self-replicators, open-ended evolution, and feasible universal construction are open problems. Asynchrony is present in the most complex natural systems such as differentiated multicellular life, but synchronous automata network models can be made asynchronous using a uniform method by which emulation of behaviour of the synchronous system is mathematically guaranteed.

**Acknowledgments.** Many thanks for their encouragement to Andreas Albrecht and the organizers of SAGA'05, and for discussions on particular areas to Aris Alissandrakis (variability in social transmission; proto-culture in artifacts), Attila Egri-Nagy (evolvability of genotype-phenotype relation in populations of self-replicating programs), R. E. Michod (discussions of individuality, sex, and fitness), John L. Rhodes (algebra and time), and Günter P. Wagner, Lee Altenberg, the late Michael Conrad (evolvability), Hiroki Sayama and Barry McMullin (self-replication and evolvability), Maria Schilstra, Wolfgang Marwan, Ren´e te Boekhorst, Kerstin Dautenhahn, Tom Ray, Hans Meinhardt, Julian F. Miller, and members of the EPSRC SEEDS research cluster. Except where stated, the ideas and results expressed in this paper, together with any shortcomings, are the author's own. Thanks to Algorithms and Adaptive Systems Research Groups, University of Hertfordshire, and the Engineering and Physical Sciences Research Council (EPSRC) for supporting the Network on Evolvability in Biological and Software Systems. The biological modelling work involving NetBuilder is supported in part by an grant from the Wellcome Trust.

## References

1. *BioSystems 69*, 2-3. Special issue on Evolvability, C. L. Nehaniv (ed.) (2003).
2. Adami, C., Ofria, C., and Collier, T. C. Evolution of biological complexity. *Proc. Natl. Acad. Sci. U.S.A. 97* (2000), 4463–4468.
3. Alissandrakis, A., Nehaniv, C. L., and Dautenhahn, K. Synchrony and perception in robotic imitation across embodiments. In *Proc. IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '03)* (2003), pp. 923–930.
4. Alissandrakis, A., Nehaniv, C. L., and Dautenhahn, K. Towards robot cultures? - learning to imitate in a robotic arm test-bed with dissimilarly embodied agents. *Interaction Studies 5*, 1 (2004), 3–44.
5. Altenberg, L. The evolution of evolvability in genetic programming. In *Advances in Genetic Programming*, K. E. Kinnear, Ed. MIT Press, 1994, pp. 47–74.
6. Arthur, W. *The Origin of Animal Body Plans: A Study in Evolutionary Developmental Biology*, 1st paperback edition ed. Cambridge, 2000.
7. Banzhaf, W. Artificial regulatory networks and genetic programming. In *Genetic Programming - Theory and Applications*. Kluwer, 2003, pp. 43–61.
8. Berners-Lee, T. Evolvability. In *7th International WWW Conference, Brisbane, Australia* (15 April 1998). keynote address: slides on-line at `http://www.w3.org/Talks/1998/0415-Evolvability/`.
9. Bonner, J. T. *The Evolution of Complexity, by Means of Natural Selection*. Princeton University Press, 1988.
10. Burks, A. W. *Essays on Cellular Automata*. University of Illinois Press, Urbana, Illinois, 1970.
11. Buss, L. W. *The Evolution of Individuality*. Princeton University Press, 1987.
12. Byl, J. Self-reproduction in small cellular automata. *Physica D 34* (1989), 295–299.
13. Codd, E. F. *Cellular Automata*. Academic Press, New York, 1968.
14. Conrad, M. The geometry of evolution. *BioSystems 24*, 2 (1990), 61–81.
15. Crutchfield, J. P. Observing complexity and the complexity of observation. In *Inside versus Outside*, H. Atmanspacher, Ed. Springer, Berlin, 1993, pp. 235–272.
16. Darwin, C. *The Origin of Species by Means of Natural Selection*, 1st ed. John Murray, London, 1859.
17. Darwin, C., and Wallace, A. On the tendency of species to form varieties; and on the perpetuation of varieties and species by natural means of selection. *Journal of the Proceedings of the Linnean Society, Zoology 3* (20 August 1858), 45–62.
18. Davidson, E. H. *Genomic Regulatory Systems: Development and Evolution*. Academic Press, 2001.

19. Dawkins, R. *The Selfish Gene*. Oxford, 1976.
20. Dawkins, R. The evolution of evolvability. In *Artificial Life*, C. Langton, Ed. Addison Wesley, 1989.
21. Dömösi, P., and Nehaniv, C. L. *Algebraic Theory of Finite Automata Networks: An Introduction (SIAM Monographs on Discrete Mathematics and Applications, Vol. 11)*. Society for Industrial and Applied Mathematics, Philadelphia, 2005.
22. Edmundson, A. C. *A Fuller Explanation: The Synergistic Geometry of R. Buckminster Fuller*. Birkhäuser, 1987.
23. Egri-Nagy, A., and Nehaniv, C. L. Evolvability of the genotype-phenotype relation in populations of self-replicating digital organisms in a tierra-like system. In *Proc. European Conference on Artificial Life (ECAL'03), September 14-17, 2003 Dortmund, Germany* (2003), vol. 2801, Springer Lecture Notes in Artificial Intelligence, pp. 238–247.
24. Fogel, L. J., Owen, A. J., and Walsh, M. J. *Artificial Intelligence through Simulated Evolution*. John Wiley, 1966.
25. Goguen, J. Requirements engineering as the reconciliation of technical and social issues. In *Requirements Engineering: Social and Technical Issues*, M. Jirotka and J. Goguen, Eds. Academic Press, 1994, pp. 165–199.
26. Goguen, J. Formality and informality in requirements engineering. In *Proceedings, Fourth International Conference on Requirements Engineering* (April 1996), IEEE Computer Society, pp. 102–108.
27. Holland, J. *Adaptation in Natural and Artificial Systems*. MIT Press, 1975.
28. Kimura, M. *The Neutral Theory of Molecular Evolution*. Cambridge Univ. Press, 1983.
29. Kirschner, M., and Gerhart, J. Evolvability. *Proc. Natl. Acad. Sci. USA 95* (1998), 8420–8427.
30. Koza, J. R. Evolution of subsumption. In *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992, ch. 13.
31. Koza, J. R. *Genetic Programming II: The Next Generation*. MIT Press, 1994.
32. Laing, R. Automaton models of reproduction by self-inspection. *Journal of Theoretical Biology 66* (1977), 437–456.
33. Langton, C. G. Self-reproduction in cellular automata. *Physica D 10* (1984), 135–144.
34. Langton, C. G. Studying artificial life with cellular automata. *Physica D 22* (1986), 120–149.
35. Lehman, M. M. The role and impact of assumptions in software development, maintenance and evolution. In *IEEE International Workshop on Software Evolvability* (2005), IEEE Computer Society Press, p. (in press).
36. Leyser, O., and Day, S. *Mechanisms in Plant Development*. Blackwell, 2003.
37. Lohn, J. D. Self-replicating systems in cellular space models. In *Mathematical and Computational Biology: Computational Morphogenesis, Hierarchical Complexity, and Digital Evolution.* Vol. 26 in Lectures on Mathematics in the Life Sciences (Providence, Rhode Island, 1999), C. L. Nehaniv, Ed., American Mathematical Society, pp. 11–30.
38. Lohn, J. D., and Reggia, J. A. Automatic discovery of self replicating structures in cellular automata. *IEEE Transactions on Evolutionary Computation 1*, 3 (1997), 165–178.
39. Lynch, M., and Katju, V. The altered evolutionary trajectories of gene duplicates. *Trends in Genetics 11* (2004), 544–549.
40. Macias, N. J., and Durbeck, L. J. K. Adaptive methods for growing electronic circuits on an imperfect synthetic matrix. *BioSystems 73* (2004), 173–204.
41. Mange, D., Sanchez, E., Stauffer, A., Tempesti, G., Marchal, P., and Piguet, C. Embryonics: A new methodology for designing field-programmable gate arrays with self-repair and self-replicating properties. In *Readings in Hardware/Software Co-Design*, G. D. Micheli, R. Ernst, and W. Wolf, Eds. Morgan Kaufmann, San Francisco, 2002, pp. 643–655.

42. MARGULIS, L. *Symbiosis in Cell Evolution.* W. H. Freeman & Co., 1981.

43. MARGULIS, L., AND SAGAN, D. *Acquiring Genomes: A Theory of the Origins of Species*. Basic Books, 2002. Foreword by Ernst Mayr.

44. MAYNARD SMITH, J. A Darwinian view of symbiosis. In *Symbiosis as a Source of Evolutionary Innovation*, L. Margulis and R. Fester, Eds. MIT Press, 1991, pp. 26–39.

45. MAYNARD SMITH, J., AND SZATHMÁRY, E. *The Major Transitions in Evolution.* W. H. Freeman, 1995.

46. MICHOD, R. E. *Eros and Evolution: A Natural Philosophy of Sex.* Addison-Wesley, 1995.

47. MICHOD, R. E. *Darwinian Dynamics: Evolutionary Transitions in Fitness and Individuality*. Princeton, 1999.

48. MICHOD, R. E., AND ROZE, D. Cooperation and conflict in the evolution of individuality. III. transitions in the unit of fitness. In *Mathematical and Computational Biology: Computational Morphogenesis, Hierarchical Complexity, and Digital Evolution.* Vol. 26 in Lectures on Mathematics in the Life Sciences (Providence, Rhode Island, 1999), American Mathematical Society, pp. 47–91.

49. MOORE, E. F. Machine models of self-reproduction. In *Proceedings of the Fourteenth Symposium on Applied Mathematics* (1962, (Reprinted in A. W. Burks (ed.), 1968)), American Mathematical Society, pp. 17–33.

50. MORITA, K., AND IMAI, K. A simple self-reproducing cellular automaton with shape-encoding mechanism. In *Artificial Life V* (1997), C. G. Langton and K. Shimohara, Eds., MIT Press, pp. 489–496.

51. NAKAMURA, K. Asynchronous cellular automata and their computational ability. *Systems, Computers, Controls 5*, 5 (1974), 58–66. translated from Japanese, *Denshi Tsushin Gakkai Ronbunshi*, **57-D**, No. 10, pp. 573–580, October 1974.

52. NEHANIV, C. L. Evolvability in biology, artifacts, and software systems. In *Proceedings of the Evolvability Workshop at the the Seventh International Conference on the Simulation and Synthesis of Living Systems (Artificial Life 7) 1-2 August 2000, Reed College, Portland, Oregon, USA* (2000). On-line at: `http://homepages.feis.herts.ac.uk/~nehaniv/al7ev/`.

53. NEHANIV, C. L. Evolution in asynchronous cellular automata. In *Artificial Life VIII: Proc. 8th Intl. Conf. on Artificial Life* (2002), R. K. Standish, M. A. Bedau, and H. A. Abbass, Eds., MIT Press, pp. 65–73.

54. NEHANIV, C. L. Internal constraints and ecology in evolution: A case study in tierra. In *Proceedings of the Fifth German Workshop on Artificial Life (GWAL V)* (2002), pp. 243–252.

55. NEHANIV, C. L. Self-reproduction in asynchronous cellular automata. In *Proc. 2002 NASA/DoD Conference on Evolvable Hardware (15-18 July 2002 – Alexandria, Virginia)* (2002), IEEE Computer Society Press, pp. 201–209.

56. NEHANIV, C. L. Asynchronous automata networks can emulate any synchronous automata network. *International Journal of Algebra & Computation 14*, 5 & 6 (2004), 719–739. Presented at International Workshop on Semigroups, Automata, and Formal Languages (June 2002 – Crema, Italy).

57. NEHANIV, C. L. The algebra of time. In *Proc. National Conf. of the Japan Society for Industrial and Applied Mathematics* (September 1993), pp. 127–128.

58. NEHANIV, C. L., AND DAUTENHAHN, K. Self-replication and reproduction: Considerations and obstacles for rigorous definitions. In *Proceedings of the Third German Workshop on Artificial Life (GWAL III)* (1998), pp. 283–290.

59. NEHANIV, C. L., AND DAUTENHAHN, K. *Artificial Life Fundamentals: The Simulation and Synthesis of Living Systems*. Springer Verlag, in prep.

60. NEHANIV, C. L., AND RHODES, J. L. On the manner in which biological complexity may grow. In *Lectures on Mathematics in the Life Sciences*, vol. 26. American Mathematical Society, 1999, pp. 93–102.
61. NEHANIV, C. L., AND RHODES, J. L. Axioms for biological complexity and mathematically rigorous measures of computational capacity: Applications to evolution of computation in cells. In *Proc. Computation in Cells: An EPSRC Emergent Computing Workshop (17-18 April 2000)* (2000), H. Bolouri and R. Paton, Eds., University of Hertfordshire, U.K., pp. 71–76.
62. NEHANIV, C. L., AND RHODES, J. L. The evolution and understanding of biological complexity from an algebraic perspective. *Artificial Life 6*, 1 (2000), 45–67.
63. OHNO, S. *Evolution by Gene Duplication*. Springer Verlag, 1970.
64. ORGEL, L. E. Molecular replication. *Nature 358* (1992), 203–209.
65. PARNAS, D. On the criteria to be used in decomposing systems into modules. *Communications of the Association for Computing Machinery 15*, 2 (1972), 1052–1058.
66. PEPPER, J. W. The evolution of evolvability in genetic linkage patterns. *BioSystems 69(2-3)*. Special issue on Evolvability, C. L. Nehaniv (ed.) (2003), 115–126.
67. QUICK, T., NEHANIV, C. L., DAUTENHAHN, K., AND ROBERTS, G. Evolving embodied genetic regulatory network-driven control systems. In *Proc. European Conference on Artificial Life (ECAL'03), Springer LNAI Vol. 2801* (2003), pp. 266–277.
68. RASMUSSEN, S., CHEN, L., DEAMER, D., KRAKAUER, D., PACKARD, N., STADLER, P., AND BEDAU, M. Transitions from nonliving to living matter. *Science 303* (2004), 963–965.
69. RAY, T. S. An approach to the synthesis of life. In *Artificial Life II*, F. Jones, Ed. Addison-Wesley, 1991, pp. 371–408.
70. RECHENBERG, I. *Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog, 1973.
71. REGGIA, J. A., ARMENTROUT, S., CHOU, H. H., AND PENG, Y. Simple systems that exhibit self-directed replication. *Science 259* (1993), 1282–1288.
72. RENDELL, L., AND WHITEHEAD, H. Culture in whales and dolphins. *Behavioral and Brain Sciences 24*, 2 (2001), 309–382.
73. RIDLEY, M. *Evolution*, 2nd ed. Blackwell Science, 1996.
74. SAPP, J. *Evolution by Association*. Oxford, 1994.
75. SAYAMA, H. *Constructing Evolutionary Systems on a Simple Deterministic Cellular Automata Space*. PhD thesis, Department of Information Science, Graduate School of Science, University of Tokyo, December 1998.
76. SAYAMA, H. Introduction of structural dissolution into Langton's self-reproducing loop. In *Artificial Life VI: Proceedings of the Sixth International Conference on Artificial Life* (1998), C. Adami, R. K. Belew, H. Kitano, and C. E. Taylor, Eds., MIT Press, pp. 114–122. On-line material at: `http://necsi.org/postdocs/sayama/sdsr/`.
77. SAYAMA, H. A new structurally dissolvable self-reproducing loop evolving in a simple cellular automata space. *Artificial Life 5*, 4 (1999), 343–365.
78. SCHILSTRA, M., AND BOLOURI, H. Logical modelling of developmental genetic regulatory networks with netbuilder. In *2nd Int. Conf. Systems Biology (ICSB 2001)*. Omnipress, 2001.
79. SCHILSTRA, M., AND NEHANIV, C. L. The logic of genetic regulation. *submitted*.
80. SCHWEFEL, H.-P. *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*. Birkhäuser, 1977.
81. SIGMUND, K. *Games of Life*. Penguin, 1995.
82. SIPPER, M. The artificial self-replication page. `http://www.cs.bgu.ac.il/∼sipper/selfrep/`.

83. SOMMERVILLE, I. *Software Engineering*, 5th ed. Addison-Wesley, 1996.

84. SZATHMÁRY, E. Chemes, genes, memes: A classification of replicators. In *Mathematical and Computational Biology: Computational Morphogenesis, Hierarchical Complexity, and Digital Evolution* (Providence, Rhode Island, 1999), C. L. Nehaniv, Ed., Lectures on Mathematics in the Life Sciences, vol. 26, American Mathematical Society, pp. 1–10.

85. SZOSTAK, J., BARTEL, D., AND LUISI, P. Synthesizing life. *Nature 409* (2001), 383–390.

86. TEMPESTI, G. A new self-reproducing cellular automaton capable of construction and computation. In *ECAL'95: Third European Conference on Artificial Life* (1995), vol. Lecture Notes in Computer Science 929, pp. 555–563.

87. TOFFOLI, T. Integration of phase-difference relations in asynchronous sequential networks. In *Automata, Languages, and Programming (Fifth Colloquium, Udine, July 1978),* Lecture Notes in Computer Science **62** (1978), G. Ausiello and C. Bohm, Eds., Springer Verlag, pp. 457–463.

88. TOFFOLI, T., AND MARGOLUS, N. *Cellular Automata Machines*. MIT Press, 1987.

89. TYRRELL, A. M., SANCHEZ, E., FLOREANO, D., TEMPESTI, G., MANGE, D., MORENO, J. M., ROSENBERG, J., AND VILLA, A. E. P. Poetic tissue: An integrated architecture for bio-inspired hardware. In *Evolvable Systems: From Biology to Hardware, 5th International Conference,ICES 2003, Trondheim, Norway, March 17-20, 2003,* Lecture Notes in Computer Science, Vol. 2606 (2003), Springer Verlag, pp. 129–140.

90. VAN BELLE, T., AND ACKLEY, D. H. Code factoring and the evolution of evolvability. In *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference* (2002), Morgan Kaufmann, pp. 1383–1390.

91. VAN BELLE, T., AND ACKLEY, D. H. Uniform subtree mutation. In *Proc. EuroGP-2002, the 5th European Conference on Genetic Programming* (2002), pp. 152–161.

92. VAN NIMWEGEN, E., CRUTCHFIELD, J. P., AND HUYNEN, M. Neutral evolution of mutational robustness. *Proc. Natl. Acad. Sci. U.S.A. 96* (1999), 9716–9720.

93. VAN NIMWEGEN, E., CRUTCHFIELD, J. P., AND MITCHELL, M. Statistical dynamics of the royal road genetic algorithm. *Theoretical Computer Science 229,* Special Issue on Evolutionary Computation, A. E. Eiben and G. Rudolph (eds.) (1999).

94. VARELA, F. J. *Principles of Biological Autonomy*. North Holland, 1979.

95. VARELA, F. J., MATURANA, H. R., AND URIBE, R. Autopoiesis: The organization of living systems. *BioSystems 5*, 4 (1974), 187–196.

96. VARSHAVSKY, V. System time and system timing. In *Algebraic Engineering*, C. L. Nehaniv and M. Ito, Eds. World Scientific Press, 1999, pp. 38–57.

97. VITÁNYI, P. M. B. Sexually reproducing cellular automata. *Mathematical Biosciences 18* (1973), 23–54.

98. VON NEUMANN, J. Probabilistic logics and the synthesis of reliable organisms from unreliable components. In *Automata Studies (Annals of Mathematics Studies, Number 34)*, C. E. Shannon and J. McCarthy, Eds. Princeton, 1956, pp. 43–98.

99. VON NEUMANN, J. *Theory of Self-Reproducing Automata.* Edited and completed by A. W. Burks. University of Illinois Press, 1966.

100. WAGNER, G. P., Ed. *The Character Concept in Evolutionary Biology*. Academic Press, 2001.

101. WAGNER, G. P., AND ALTENBERG, L. Complex adaptations and the evolution of evolvability. *Evolution 50*, 3 (1996), 967–976.

102. WATSON, J. D., AND CRICK, F. H. C. Molecular structure of nucleic acids. *Nature 171* (1953), 737–738.

103. WERNICK, P., AND LEHMAN, M. M. Software process white box modelling for FEAST/1. *Journal of Systems and Software 46*, 2-3 (1999), 193–201.

104. WEST-EBERHARD, M. *Developmental Plasticity and Evolution*. Oxford University Press, 2003.

105. WHITEN, A., GOODALL, J., MCGREW, W. C., NISHIDA, T., REYNOLDS, V., SUGIYAMA, Y., TUTIN, C. E. G., WRANGHAM, R. W., AND BOESCH, C. Culture in chimpanzees. *Nature 399* (1999), 682–685.
106. WOLPERT, L. *The Triumph of the Embryo.* Oxford University Press, 1991.
107. WRIGHT, S. The role of mutation, inbreeding, crossbreeding, and selection in evolution. *Proceedings of the Sixth International Congress on Genetics 1* (1932), 356–366.

## Appendix A: Are There Degrees of Life? – Converting Resources into Persistence and Progeny

The preceding considerations of evolutionary processes as stochastic algorithms that may be realized in many substrates lead naturally to the question of when the individuals in an evolutionary process should be called *alive*. We consider in this appendix what heritability, self-maintenance, symbiosis and responsive dynamics via genetic regulatory networks can tell us about *possible modes of life,* in whatever medium it might find realization.

### A.1 Re-Thinking Life

The last two centuries have yielded profound scientific advances in our understanding of the particular nature of life on earth: the nature of the cell, the Darwinian theory of evolution, its synthesis with Mendelian genetics, and later with biochemistry, the basis of hereditary in the substance of long chains of nucleotides, the (nearly) universal genetic code, the details of protein biosynthesis (see e.g. [73]), increased understanding of the dynamics of development in animals [6] and plants [36], and of evolvability [1, 100], as well as ever more detailed understanding of genetic regulatory systems and how they operate in the single-celled and differentiated multicellular organisms (e.g. [18]), among other advances. Recent advances in the construction of 'proto-cells' with various properties of living systems bring us closer to new and very minimal instantiations of life – of various kinds (artificially constructed or parred-down cells) [68, 85]. We may possibly find other examples of life elsewhere in the solar system or universe, but also in media other than those of organic biochemistry. Researchers in Artificial Life have sought to capture and reproduce an underlying an "logic of life" in software running on electronic computers (e.g. [69]), or in other media. These developments are leading us to reconsider the notion of living organism: How are we to know whether we are looking at new kinds of life?

**A.2 Some Subtle Properties of Life** Here we collect fundamental phenomena found in organic life on earth that have tended not to receive emphasis. We formulate them with a view toward achieving a more universal understanding of what life is.

**Replicators in Context.** The heritable material in replicators does not fully specify the constraints of the environment in which they are capable of replication. Indeed, there is no way for evolution directly to distinguish between genetic, environmental, physical, geometric, energetic constraints, or incidental or universal properties of an evolving population's environment. Catalyzing the ambient dynamics, no matter how, that promote their own replication is enough for the most primitive replicators. Other capabilities come later in the evolution of biological complexity [62].

**Pattern Integrity** Life is an organizational pattern that "holds its shape". Although the particular material that makes up an organism in the course of its life may be changing, the individual persists. Just as a wave on the ocean, a dynamical event occurring on a substrate of water molecules, and a slipknot, which may be passed down from a string to a rope to a necktie, depend on *configuration* and not particular material for their existence, similarly life relies on persistence of an organizational pattern that is structurally stable enough to perpetuate its existence in a medium (*pattern integrity*). (cf. discussion in [22].)

Enough stability must be present in the structure of the dynamics for this to occur for us to be able to speak about an *individual* of any kind.

In replication, during a transition when a parent is giving rise to its offspring in the dynamics of the medium, the parent is active in the establishment of the offspring's pattern integrity, as is the new individual itself, in changing degrees (*self-production*). Replicators whose dynamics favors conditions that happen to tend to increase their reproductive success through promoting processes of homeostasis and self-repair will increase under the action of evolution. Beyond basic pattern integrity – which is necessarily present already with simple replicators, or for that matter in any persistent individuals – life has also achieved more sophisticated types of pattern integrity in the forms of *self-maintenance*, *homeostasis*, *self-repair* and *regeneration* within an individual.

**Robustness, Plasticity and Incessant Responsiveness** Despite perturbation and variability at the level of inheritance (such as mutations and sex), living organisms are often robust and can grow, develop, persist, and thrive successfully (*robustness to genetic variability* [14]). Moreover, despite the lack of any full 'specification' of organisms by their genetic material and despite perturbations to the environment, they are often remarkably able to generate appropriate and adaptive forms and responses to various, changing environmental situations (*phenotypic and developmental plasticity*) [104]. This relies on the fact that organisms continually engage with their local environment, showing incessant adaptive activity, internally and externally (*universal responsiveness* [104]).

**Evolvable Dynamical Systems.** *Evolvability* (cf. [1]) is defined above as the particular capacity of an evolving population to generate adaptive heritable genotypic and phenotypic variation. Evolvability depends on many details of the genetic system and can be radically different in different instances of evolution. To achieve self-maintenance, plasticity, and continual responsiveness a sufficiently evolvable substrate is necessary. Sufficiently powerful dynamical systems bases (such as the physics, chemistry and genetic system enjoyed by life on earth) are necessary to support evolvability. Moreover, the evolvability of a system can change (e.g. with the advent of a genetic code).

**Achieving Heritability and Transitions in Individuality.** Things fall apart (in the physical and many artificial worlds). Quick reproduction based on *digital templates* is one of the ways that life uses to help it to triumph over entropy. With a discrete basis of heritability (such as provided by a limited alphabet of nucleotides and codons), copies can be perfect. Engineering shows that rebuilding is often more efficient than repair.

Simple replicators might have no regeneration and repair capabilities (and these might be quite complex to specify in a heritable manner). Without these, replication rather than repair is eventually the only option in the face of the tendency of things to crumble. With increased degrees of self-repair and regeneration, *persistence* rather than immediate production of offspring becomes an option. Too much persistence however would stop evolution, unless variability is somehow continually generated.

The simplest replicators have offspring like themselves in a similar environment. Fitness of these is likely to be similar to that of the parents (until resources are exhausted, or conditions change): When offspring are produced, they are likely to be successful in the environment of their parent if they are similar to it – that is, if their capabilities of promoting their own persistence and reproductive success in that environment are similar to those of their parents. Heritability of fitness is achieved simply for accurate replicators in a stable environment: if the new one is to be viable, a copy of the original is likely to be a good choice.

Ensuring faithfulness of copies is one way to ensure heritability of fitness. Digital genetic systems help achieve this.

New higher-level, e.g. multicellular, replicators whose components are themselves alive need to solve the problems of heritability of fitness anew, together with those of sex, and self-repair. At higher levels, for life to exist, a transition to individuality must reinvent replication, resulting in heritability of fitness at that level; also reinvented are self-maintenance for the higher-level individual, and sex (receiving or exchanging of heritable material from others) [47]. It is also faced with new problems: suppression of freedom at lower levels and harnessing lower level units into cooperatives that contribute to the fitness of the higher level individual requires a balance of the tendency of replicators at lower levels to pursue their own individual reproductive success at the expense of that of the higher level individual [11, 47]. In differentiated multicellularity, constituent cells may pursue their own replication at the higher-level's expense (cancer); in social insects some members of non-reproductive labor castes may 'defect' and become reproductive, at the expense of the colony's integrity.

Higher-level individuals develop from a single (or a small number of) constituents and harness division of labor and the differentiation of their constituents (e.g. cells in a body; insect castes). Epigenetic inheritance (via state and marking) in the population of constituents comprising them makes this possible.

### A.3 Degrees and Modes of Being Alive

Biologists asked to define life have not agreed on a universally accepted definition, but instead tend to produce lists of properties. One candidate definition [59], applicable to life-as-it-could-be as well as life-as-we-know-it, is:
A *living organism* is an individual entity that

1. transforms resources into persistence (pattern integrity) and progeny (i.e. to achieve reproduction),
2. results from reproduction (or is able to reproduce),
3. results from an evolutionary process in a population (involving heritable variation, differential reproductive success, finite resources),

4. produces itself in the context of its environment (self-production, growth and possibly development),
5. engages in self-maintenance,
6. uses signaling and interaction (internally and externally), and
7. modifies and is modified by its environment (embodiment and plasticity).

The more of these properties from such lists a system exhibits, the more justified we feel in saying that it is alive. For example, viruses or ‘digital organisms’ (such as Tierrans [69]) exhibit some but not all the properties of life (see also below). This suggests that there may be different degrees to which it makes sense to call something alive, i.e. different *degrees of life* depending on the degree to which each of requirements in the biologist’s list are satisfied [59].

**Modes of Life with Different Degrees of Aliveness** In light of the advances in Biology and Artificial Life mentioned above, it will be argued here that the evolution of life from a single ancestor or ancestral population may involve different modes which account for some of the differences in the degree of aliveness. Organic evolution on earth (and in some cases in artificial systems at the first two levels) has apparently produce different *modes of life* (which comprise a evolutionary continuum). These modes show marked qualitative differences in dimensions present among the requirements for life, but are related to each other by evolutionary contingencies:

- Mode 0: **Replicators (no or very limited variation in heritability).** Prions, crystal growth, and cellular automata replicators show many properties of life. They engage in no self-maintenance. Evolution does not occur (with some recent exceptions in cellular automata). At the level of the individual, they do not show genetic, developmental, or phenotypic plasticity.
- Mode 1: **Replicators in Evolving Populations.** Viruses, Tierrans (and to a lesser degree transposons) exist in populations showing heritability, variability, and differential reproductive success – the requirements for evolution. This leads some researchers to assert that they are examples of minimal living systems. They exhibit no homeostatic control, or capabilities of regeneration and self-repair.
- Mode 2: **Self-Maintaining Organisms (showing various degrees of autonomous responsive dynamics and plasticity).** These extend mode 1 capacities and include all the uncontroversial cases of living organisms. [See list of *living organism* dimensional properties above. This class might have identifiable occurring submodes, e.g. some self-maintenance and regenerative capability but little or no phenotypic plasticity. Some examples may be naturally occurring, but others might soon be produced by constructive methods en route to more sophisticated artificial life forms, e.g. proto-cells.]
- Mode 3: **Degenerated Life (arising, e.g., due to Symbiogenesis or Multicellularity).** These are evolving replicators whose ancestors were of another mode but which are in the process of losing (or have lost) most of their individuality. Examples associated with evolutionary transitions: in the RNA world (or other early life scenarios), early replicators → membrane-bound genes/‘chromosomes’; in the evolution of eukaryotes, free-living prokaryotic ancestors of organelles → mitochondria, chloroplasts. Mitochondria and chloroplasts are organelles whose ancestors

where free-living endosymbionts of ancestors of eukaryotic cells but which are not longer capable of replication without the machinery of the cells and some of the heritable information in the cell's nuclear DNA [42, 74].

**Transitions between Modes of Life** The advent of new higher-level replicators such as multicellular entities (possibly with differentiation of constituents into a cooperative division of labor) leads us to ask the question again. Are these replicators (rather than their constituent members) alive? To what degree do they show the properties required of living systems such as being members of an evolving population? self-maintenance, heritability of fitness, and sophisticated responsive dynamics?

If multicellular plants, animals and fungi, or colonies of social insects are self-replicating higher level individuals satisfying the properties of life, how are the properties of life (in the list above) achieved by the higher level entity? Are we dealing with mere replicators, evolution, or evolution of self-maintaining dynamic entities?

With symbiogenesis and increasing dependency on partners, loss of individuality may result (e.g. in the evolution of cellular organelles having endosymbiotic origin, mitochondria and chloroplasts). For most definitions of life, evolution must act on individuals in a population, if individuality is lost, then evolution at the former level of individuality – and hence life at that level – becomes less distinct (mode 3), and eventually it may not be appropriate to speak of life.

The extreme modes are less alive than the middle ones. Transitions between the above modes of life (in the direction of the list) are however natural and may be favored by natural selection (as with the evolution of mitochondria). Even though the last mode involves degeneration of life properties at one level, it is only known to occur in the transitional genesis of higher-level units of selection. Note that it also need not occur (even with new, albeit loose units of selection); e.g. the cells in differentiated multicellular organisms or symbiotic partners in lichens that can also live independently are both mode 2. In the opposite direction, examples which may or may not have arisen as renegade replicators that were originally components of a larger unit of life include viruses and transposons (transitions to mode 1 from higher modes).