Visualizing Evolutionary Dynamics of Self-Replicators: A Graph-Based Approach

Chris Salzberg
Department of Human Communication
University of Electro-Communications
1-5-1 Chofugaoka, Chofu
Tokyo 182-8585, Japan
and
Graduate School of Arts and Sciences
University of Tokyo
3-8-1 Komaba, Meguro
Tokyo 153-8902, Japan
chris@sacral.c.u-tokyo.ac.jp

Antony Antony
Section Computational Science
Universiteit van Amsterdam
Kruislaan 403
1098 SJ Amsterdam, The Netherlands
antony@phenome.org

Hiroki Sayama*
Department of Human Communication
University of Electro-Communications
1-5-1 Chofugaoka, Chofu
Tokyo 182-8585, Japan
and
Department of Bioengineering
Binghamton University, State University of New York
P.O. Box 6000
Binghamton, NY, 13902
sayama@binghamton.edu

Abstract We present a general approach for evaluating and visualizing evolutionary dynamics of self-replicators using a graph-based representation for genealogy. Through a transformation from the space of species and mutations to the space of nodes and links, evolutionary dynamics are understood as a flow in graph space. A formalism is introduced to quantify such genealogical flows in terms of the complete history of localized evolutionary events recorded at the finest level of detail. Represented in a multidimensional viewing space, collective dynamical properties of an evolving genealogy are characterized in the form of aggregate flows. We demonstrate the effectiveness of this approach by using it to compare the evolutionary exploration behavior of self-replicating loops under two different environmental settings.

Keywords
Visualization, genealogy graph, evolutionary dynamics, self-replication

1 Introduction

Research on artificial self-replication has resulted in a variety of systems exhibiting complex evolutionary dynamics [17]. Of crucial interest in the analysis of these systems are the localized

* Corresponding author.

events (interactions, mutations) that collectively decide the path of global trends in evolution. Few attempts have been made to visualize the topology of this transition space in a general way. The method proposed by Bedau and Brown [1], for example, visualizes a genotype's evolutionary activity by integrating its concentration in the evolving population over time. This method implicitly circumvents the problem of tracing lineages and hence provides no clues about transition space topology. While useful as a global indicator, this and other such frequency-based approaches overlook the very fluctuations that enable evolution to occur: These are the genealogical *links* relating distinct species through their ancestry. Without these links, a gap remains between the global dynamics we observe and the localized interactions that trigger their emergence.

In this article we attempt to bridge this gap. As an alternative to earlier methods of analysis, we approach the visualization problem from the level of genealogical transitions rather than genotypical frequencies. We do so by introducing a transformation from the space of self-replicator species and their mutations to an abstract graph space where nodes and links represent species and mutations, respectively. The reproductive activity of each replicator species is quantified in this space by its *production,* a metric reflecting the frequencies of outgoing, incoming, and self link traversals associated to its corresponding node in the new graph space over a given time window. Species are then classified according to their production on a scale between *source* and *sink*; source species are typically self-replicative and often produce other species through mutations, continuously introducing active individuals into the population, whereas sink species are produced predominantly by other species. In this picture, the temporal evolution of populations and their genealogical connectivity are conceptualized as a *flow* from source to sink nodes.

To demonstrate our approach, we present an example of graph-based genealogy visualization applied to a simple self-replicating cellular automaton, the Evoloop [15]. We highlight the effectiveness of our approach by comparing the evolutionary dynamics of loops under two different environmental settings: one with the original state transition rules and one with a newly introduced *pathogen* state [13]. The proposed approach using graph-based genealogy successfully captures the critical role played by dynamic habitat partitioning in the maintenance of evolutionary exploration behavior.

Work related to the approach presented here has been performed by a number of groups. Schuster and Fontana [16] focus on adaptive trajectories of RNA sequences through RNA "shape" space using the concept of a "relay series." A key distinction is that the relay series are target-oriented; gene sequences that do not form part of the relay series are not visualized. This is not true of the method presented here. Lenski et al. [7] explore the evolutionary origin of complex features. Genealogical analysis in this work is also exact; there are no "missing links" in evolutionary paths. Yet the focus in [7] is on functional genomics for the Avida model rather than evolutionary dynamics for a general system, which is the emphasis of our work.

2 Graph-Based Genealogy Analysis

We derive a general, graph-based picture of evolution from a set of basic definitions. In what follows we limit ourselves to asexually reproducing systems where reproduction occurs via binary fission.

2.1 Definitions

Our approach assumes an arbitrary system of self-replicators evolving over time in a spatial domain $\mathbf{C}$, which we call *configuration space*. This may be a cellular automata space [6, 15], the memory and CPU of a computer [8, 9, 19], or any other well-defined domain. The intrinsic structure of individual self-replicators in this domain is described uniquely by a sequence of digits from an arbitrary alphabet, which we call an *identifier*. An identifier may be a gene sequence, a program code, or any other well-defined modular form that specifies the replicator's evolutionary identity, possibly including phenotypical information. We call the space of these identifiers $\mathbf{\Theta}$. For the purposes of this article we will assume $\mathbf{\Theta}$ to be a countably infinite set. In principle, however, uncountably infinite (e.g., real-valued) identifier sets are also possible.

To each replicator in configuration space we associate a position $r \in \boldsymbol{C}$ and identifier $\theta \in \boldsymbol{\Theta}$; the pair (r, θ) uniquely describes an *instance*. The creation of a new replicator is defined as a *birth* and is described by parent θ_p, child θ_c, time of birth t_b, and location of birth r_b. A birth for which $\theta_p \neq \theta_c$ indicates that mutation has occurred. Following Bedau and Brown [1], we define the time evolution of birth events over the domain $\mathbf{C}$ by the birth *trigger* function δ_b:

$$\delta_b(\theta_p, \theta_c, t_b, r_b) = \begin{cases} 1 & \text{if a new replicator } \theta_c \text{ is born from a parent } \theta_p \\ & \quad \text{at time } t_b \text{ at position } r_b, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

A *death* event—generally associated with the cessation of self-reproductive capability—is described in a similar manner, but by a single identifier θ_d. We define the death trigger function δ_d for death events as

$$\delta_d(\theta_d, t_d, r_d) = \begin{cases} 1 & \text{if a replicator } \theta_d \text{ died at time } t_d \text{ at position } r_d, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

In what follows, we use the functions δ_b and δ_d as our fundamental quantities. Note that most existing analysis of artificial models implicitly assume a tree-based genealogy [9, 15, 19]; Equations 1 and 2 make no such assumption (see Section 2.2, below). We do however assume that we may identify parents and children in a unique and unambiguous manner, and that births may be precisely defined in both time and space. While practically impossible in real biological systems, such complete data collection can be done for artificial evolutionary models.

To complete our set of definitions, we introduce a partitioning of the space of identifiers $\boldsymbol{\Theta}$ into countably many nonempty sets; these are the entities we call *species* in this article. Each species set, denoted by a unique index used for this purpose only, contains one or more identifiers considered of the same type according to some well-defined classification criteria. The definition of boundaries between these species sets is problem-specific and may in principle be arbitrarily specified by the system observer. One possibility would be to group multiple identifiers that share the same phenotypical characteristic (if the system allows many-to-one mappings from genotype to phenotype). The earlier size-based analysis of Evoloops [15] is a typical example of such classification.

Note that the choice of species sets has important consequences for the structure of the genealogy space to be visualized. In particular, the volume occupied by each species in $\boldsymbol{\Theta}$ determines the resolution of the analysis and its robustness to genetic drift: it is only the *interspecific* mutations—and not *intraspecific* ones—that have an influence on the results of analysis. If each unique identifier is considered a distinct species, the analysis is at its highest resolution; any mutation, regardless of its type, is detected and expressed in the classification as the formation of a new species. In Section 3 we use this high-resolution classification to demonstrate how our approach works.

2.2 Genealogy on a Graph

The basis for our genealogy analysis is a transformation from configuration space $\mathbf{C}$ to a directed graph $\mathbf{G}$, which we call a *genealogy graph*. Although not strictly necessary, we assume time and space to be discrete hereafter. Definitions describing this graph cover a time window from t_i to t_f, which is written as $\mathbf{T} = [t_i, t_f)$. We assign a node in the graph for each species, and associate it with a unique index k and an initial population $\mathbf{P}(k, t_i)$. We write species k as a set of identifiers, $S_k = \{\theta_1^k, \theta_2^k, \ldots\}$. Directed edges in this graph represent ancestral links created with the birth of replicators: Following

detection and identification, a parent node k is assigned to the newborn node l. The pair of species (k, l) are henceforth distinguished as parent and child relative to the directed link between them.

Note that nodes in **G** represent *groups*, not *instances*, of self-replicators. Here we deal with systems of what Szathmáry calls "limited heredity replicators" [18], for which the number of possible different types is about equal to or smaller than their population size, so that the same type may be realized many times during evolutionary exploration processes. Such *evolutionary cycles*, easily described in terms of graph-based genealogy, have no representation in traditional tree-based structures. Nodes in such graphs may have multiple incoming links corresponding to mutations undergone by instances of several distinct species. The frequency of these repeat events is dependent on the relative size of volumes occupied by distinct species in the identifier space **Θ**. For systems where each interspecific transition always creates a novel, unique species, each node has one (and only one) incoming link, so that conventional tree-based genealogy becomes more efficient. The graph-based genealogy analysis we propose, however, includes tree-based genealogy as a special case; thus the formulations discussed below apply, with no modifications, for these kinds of systems as well.

To track the evolution of graph-based genealogy, we introduce a traversal frequency function F(k, l, **T**) describing the number of link traversals (births) from node k (parent) to node l (child) in the interval **T**, derived from δ_b as

$$\mathrm{F}(k, l, \mathbf{T}) = \sum_{\theta_p \in S_k} \sum_{\theta_c \in S_l} \sum_{t \in \mathbf{T}} \sum_{r \in C} \delta_b(\theta_p, \theta_c, t, r). \tag{3}$$

From the traversal frequency, we derive three important quantities: the number of incoming link traversals I(k, **T**), the number of outgoing link traversals O(k, **T**), and the number of buckle (self-link) traversals B(k, **T**) in the interval **T**:

$$\mathrm{I}(k, \mathbf{T}) = \sum_{l \neq k} \mathrm{F}(l, k, \mathbf{T}), \tag{4}$$

$$\mathrm{O}(k, \mathbf{T}) = \sum_{l \neq k} \mathrm{F}(k, l, \mathbf{T}), \tag{5}$$

$$\mathrm{B}(k, \mathbf{T}) = \mathrm{F}(k, k, \mathbf{T}). \tag{6}$$

To complete the dynamics of our system, we define a counter representing the number of deaths of a species k in the interval **T** as D(k, **T**):

$$\mathrm{D}(k, \mathbf{T}) = \sum_{\theta_d \in S_k} \sum_{t \in \mathbf{T}} \sum_{r \in C} \delta_d(\theta_d, t, r). \tag{7}$$

The above equations can also apply to other settings with continuous space and/or time; in this case, relevant summations are replaced by integrals with δ_b and δ_d converted to superpositions of Dirac delta functions along continuous axes. These quantities allow us to derive the change in population of the species k over **T** as

$$\mathrm{P}(k, t_f) = \mathrm{P}(k, t_i) + \mathrm{I}(k, \mathbf{T}) + \mathrm{B}(k, \mathbf{T}) - \mathrm{D}(k, \mathbf{T}). \tag{8}$$

Note that the number of outgoing link traversals $O(k, \mathbf{T})$ plays no direct role in the population change of species k; these traversals (replication events) affect only the populations of the destination species.

In transforming from δ_b and δ_d to $F(k, l, \mathbf{T})$ and $D(k, \mathbf{T})$, information about spatial distribution and details of replicator identities has been lost. However, the population dynamics and genealogy for our system—the quantities of interest for our analysis—are still completely described. Moreover, the emphasis in this formalism is on the dynamics of evolutionary connectivity rather than on global cumulative trends. The advantage to such an approach is that we retain statistical properties of individual species within the context of their ancestral links; these links play a critical role in the emergent phenomena we observe.

2.3 Evolution as Flow

The framework defined above inspires a change in the way we understand the dynamics of evolution. The alternative we propose here is the idea of *evolution as flow*. In this picture, an evolutionary system is composed of a subset of nodes in genealogical graph space, each of which represents a distinct type of replicator to which a population and collection of active incoming and outgoing links are attributed. Self-replication and variation correspond to the traversals of self-links and outgoing links in graph space, respectively, which induce a collective motion of the global population.

To reflect the above ideas in terms of a *flow*, we derive a quantity that we call the *production*:

$$\mathrm{Prod}(k, \mathbf{T}) = \mathrm{O}(k, \mathbf{T}) + \mathrm{B}(k, \mathbf{T}) - \mathrm{I}(k, \mathbf{T}). \tag{9}$$

According to this definition, replicator species that frequently construct individuals of other species as well as individuals of their own species will have a high production; those that are frequently constructed by other species but fail to self-replicate will have negative production. Note that the production is not the same as the fitness of a species; in addition to self-reproduction (B), it also considers the existence of outgoing and incoming links (O and I) and thus emphasizes genealogical connectivity. Borrowing terminology from physics and network theory, we describe a node with highly positive production as an evolutionary *source*, and a node with highly negative production as an evolutionary *sink*. Ranking nodes in this way quantifies the role species play in the evolutionary process. The balance in Equation 9 is hence between the capability of a species to *produce* and its tendency to *be produced*.

For the purposes of this article, visualization of this evolutionary flow in a genealogy graph is represented in the form of a sequence of *snapshots* for each time window $\mathbf{T}$ mapped to a multidimensional viewing space. The directionality of links in these snapshots are inferred from the production values of source and target nodes (i.e., from the flow) rather than from the links themselves. In this way, large and complex genealogies may be visualized in their entirety as undirected graphs with weighted nodes and links. The aim of this approach is to characterize collective dynamical properties of an evolving genealogy in the form of aggregate flows, exposing patterns in genealogical transition space topologies.

In principle, the scheme used to translate graph nodes (species) and links to a visualization space may be arbitrarily defined, depending either on details of identifier structure specific to the model under consideration (as in, e.g., [11]) or purely on graph space topology (as in, e.g., Section 3.2). The example we describe in the next section uses an iterative relaxation scheme to map the genealogy of a system of self-replicating loops to a two-dimensional space. This example is presented as a starting point for implementing the ideas we have presented so far.

3 Example: Evoloop with Pathogenic Environment

In this section we apply the proposed method to a simple self-replicating cellular automaton, the Evoloop [15]. The Evoloop satisfies the three conditions necessary for evolution to occur:

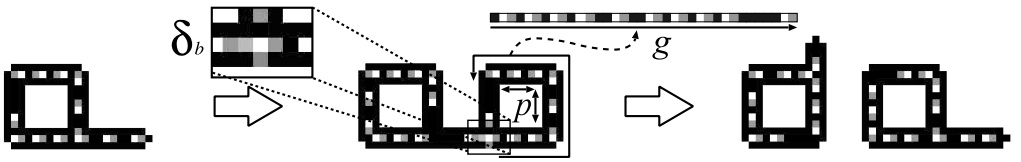

Figure 1. Self-replication of the Evoloop. g is the genotype, p the phenotype (length and width), and δ_b (middle frame) the configuration that triggers a birth event.

replication, variation (mutation), and differential fitness (competition) [2]. This model is also simple and scalable, taking the form of nine-state cellular automata with a von Neumann neighborhood. Structurally, the Evoloop is divisible into two basic components: an inner and outer sheath of square or rectangular shape, and a gene sequence of moving signal states. Coordination of the duplication process is controlled via the sequence of genes within the external sheath; mutations occur through extrinsic interaction, leading to a change in the gene sequence of offspring loops. This results in a uniquely emergent process of evolution, which—due to their robustness and high replication rate—generally favors smaller-size loops [15].[1] In what follows, definitions discussed in Section 2.1 are linked to model-specific events governing self-replication in the Evoloop CA. We then use an iterative relaxation scheme to map the nodes and links of its genealogy graphs to a 2D visualization space, purely on the basis of genealogy-space topology. The introduction of a pathogenic environment [13] is used to demonstrate the insight gained by this approach.

3.1 Defining Trigger Functions and Species

Our analysis begins with a model-specific definition for *birth*. We use for this purpose what we call the *umbilical cord dissolver*, a state that appears upon closure of the arm and then retracts towards the parent loop. The local configuration highlighted in the second frame of Figure 1 signals that such an event has occurred. At birth, loops are assigned a *genotype* $g \in \Gamma_g$ corresponding to the configuration of genes in their gene sequence read counterclockwise from the location of the umbilical cord dissolver, and a *phenotype* $p \in \Gamma_p$ describing their size (length and width). The sequence of genes in g is expressed in a compact format in which states are grouped based on dynamic properties of the CA rules. The resulting sequence has an alphabet consisting of three elements: G genes (071) controlling growth of the arm, T genes (041) controlling left turning of the arm, and free C states (1) that fill gaps in the sequence. The space of identifiers, Θ, is the space of all possible combinations of genotype and phenotype; hence $\Theta = \Gamma_g \times \Gamma_p$ (with a constraint that the phenotype must be large enough to contain the accompanying genotype). Each $\theta \in \Theta$ contains the necessary information to reconstruct a loop in the exact configuration it was in when it was born. For the birth event in the middle frame of Figure 1, identification reveals that g = GGGGCGCGTTGCCCCG and $p = 8\times8$; hence the newborn loop has GGGGCGCGTTGCCCCG/8 × 8 as its identity. Given the time t_b of the middle frame of Figure 1 and the location r_b of the umbilical cord dissolver, the birth trigger function $\delta_b(\theta_p, \theta_c, t_b, r_b)$ now has a precise meaning for the Evoloop model. In an analogous manner, we link the death trigger function δ_d to the disappearance of any inner sheath state of a loop.[2] Finally, we consider each distinct identifier a separate species in what follows, that is, $S_k = \{\theta_k\}$ for all k. This choice is made in order to simplify the analysis and to demonstrate the finest possible resolution of our approach. With δ_b, δ_d, and species defined, the graph-based parameters derived in Section 2.2 are now fully described.

3.2 Mapping to a 2D Space

Different mapping schemes have been used in the past to visualize graph-based evolutionary dynamics for the Evoloop system. To emphasize the generality of our approach, we apply in this

1 Deviations from exclusively size-based fitness selection, however, have recently been found for a subclass of replicator species [10, 12, 14].

2 For a more detailed implementation description, see [10].

article a dynamic mapping that ignores the specifics of identifier structure described above, instead focusing exclusively on the topology of genealogy space; this differs from the heuristic mapping scheme presented in earlier literature [11]. We use an iterative relaxation method that treats a collection of nodes and links as a dynamical system whose minimum potential states correspond to visually meaningful configurations in a 2D space. Two criteria are imposed: that nodes with highly traversed links should be close to each other in the visualization space, and that each node should maintain a minimum distance between itself and other nodes. Based on these requirements, our iterative relaxation scheme transforms a collection of nodes and links from an initially random configuration to a visually insightful one, through pseudo-physical simulation of motions of nodes in the 2D visualization space. Details of implementation are included in the Appendix. We note, however, that our focus in the following section is on the general approach of graph-based genealogy visualization, rather than specifics of the mapping scheme used.

To make it easy to visually detect temporal changes of the genealogy graph, we first apply the above-mentioned iterative relaxation scheme to traversal frequency data collected for the entire length of the simulation run, and then draw a successive sequence of graphs, one for each window $\mathbf{T}$, using the fixed coordinates for each species calculated by the preceding relaxation. The following representations are used for drawing species in the 2D visualization plane:

- Evolutionary sources ($\mathrm{Prod}(k, \mathbf{T}) > 0$) are marked with blue circles.
- Evolutionary sinks ($\mathrm{Prod}(k, \mathbf{T}) < 0$) are marked with red triangles.
- Sizes of markers for sources and sinks are scaled relative to the magnitude $|\mathrm{Prod}(k, \mathbf{T})|$.
- Links between species are represented by green lines whose thickness is determined by the cumulative traversal frequency over the time window $\mathbf{T}$. Links with extremely frequent traversals are distinguished with black. Arrowheads are omitted here to make the plots concise. In the case of asymmetric bidirectional links, the thickness of the higher-frequency direction is used.

Again, note that these specifics of representation are not our main focus in this article. They constitute our first attempt to graphically plot the results of graph-based genealogy analysis, and may benefit from supplementary visual cues such as small arrowheads or tapered links for improved intelligibility.

3.3 Results

Experiments were run on an 800×800 grid with periodic boundary conditions applied to its edges, beginning with a single loop of species GGCCGGCGCGTTGCGC/8×8 (denoted as *a*). The population dynamics, plotted in Figure 2, demonstrate a rapid convergence to small-size species within the first 30,000 iterations, at which point the system settles into an attractor dominated mainly by three distinct size-4 species (denoted as *b*–*d*) with roughly equal fitness.

Genealogy visualization is shown in Figure 3, with data binned over periods of 10,000 iterations. The observed evolution towards small-size species is predicted by the results of Figure 2, but Figure 3 also reveals the topology of the genealogical transition space leading to this outcome. Evolutionary exploration manifests itself as an initial expansive cascade of new source nodes in the period $\mathbf{T} = [10000, 20000)$, followed by exclusionary domination in the period $\mathbf{T} = [20000, 30000)$, and finally stagnation from roughly 30,000 onwards. Note that, by including sets of transition links in their entirety, Figure 3 expresses not only the importance of dominant species but also the shape of the exploration process occurring throughout the population. The many kinks in transition paths indicate the presence of species that are neither source nor sink. These *transient* species produce a different species than their own, playing the role of intermediary between stable species in the process of evolution; this can be understood as the active mutation discussed by Ikegami [5]. Due to their low numbers, many traditional methods of

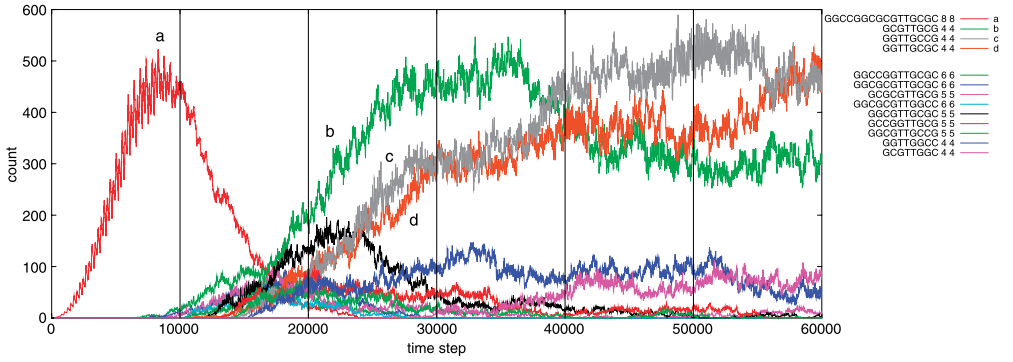

Figure 2. Population dynamics of the original Evoloop system. The space used is an 800 × 800 grid with periodic boundary conditions. The ancestral individual is of species GGCGGCGCGTTGCGC/8 × 8. Counts are plotted only for species whose population exceeds 50 at any time during the simulation run. Dashed vertical lines show divisions between time windows used in Figure 3.

population analysis would miss these species, yet they play a crucial role in shaping the evolution of populations.

To illustrate the effectiveness of the proposed visualization method, we compare the above result with evolutionary dynamics in the presence of the pathogenic environment introduced in [13]. This environment consists of a newly added tenth state that we call *pathogen*, as well as an added CA layer. The new pathogen state spreads over contiguous loop structures upon contact and then remains in the CA space for a specified period of time, its lifetime tracked at each location by counters in the added layer. A pathogen cell that has surpassed its lifetime—given as a global parameter to the system—is removed from the space (i.e., its state is changed to the quiescent state 0). Note that this new state is strictly part of the environment and thus is in no way integral to

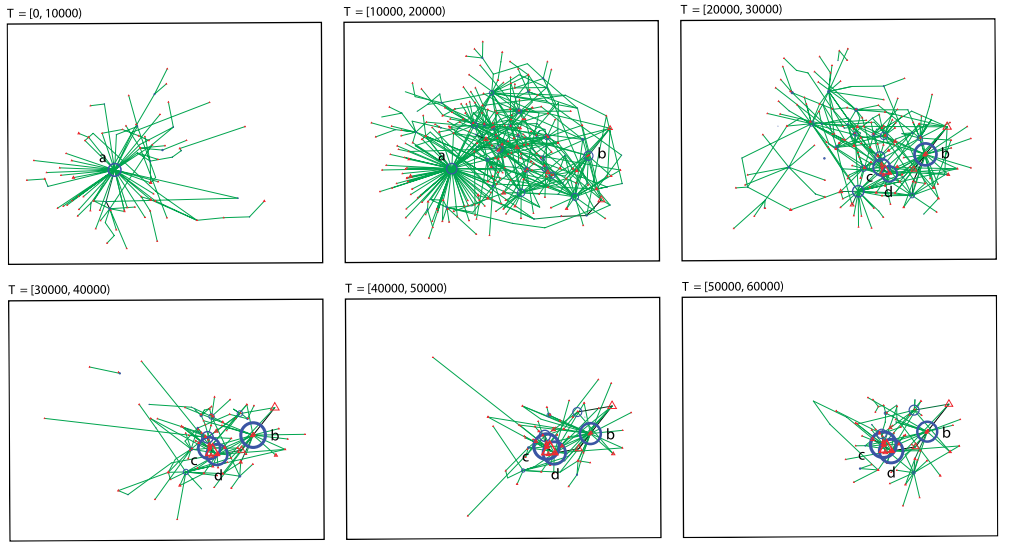

Figure 3. Visualization of the evolutionary dynamics shown in Figure 2 using graph-based genealogy analysis. The sequence of images reads rowwise from left to right, with each image covering 10,000 iterations from $T = [0, 10000)$ to $T = [50000, 60000)$. The x and y coordinates of each node were calculated from the relaxation scheme described in the Appendix.

the basic functionality of the original Evoloop CA; loops will never produce it on their own, so at least one cell in the initial configuration must be in the pathogen state to introduce the pathogenic environment.

The basic behavior of the pathogen is a wavelike propagation over susceptible populations, similar to those reported in spatial predator-prey or host-parasite systems. The effect of this new environment is twofold: to open free space and to dynamically partition the periodic domain, as seen in Figure 4. The combination of these effects has been found to encourage sustainable diversity, leading to evolutionary accessibility to larger species and a limited form of complexity increase [13].

Figures 5 and 6 depict population dynamics and genealogy exploration starting with the same ancestral species (GGCCGGCGCGTTGCGC/8 × 8), now coupled with the pathogenic environment. From Figure 5, we observe an initial domination by the size-8 species, followed by competition with a number of different species of both smaller and larger sizes (denoted as *b*–*f*). It is clear that the evolutionary dynamics in this figure are different from those shown in Figure 2, as evidenced for instance by the variety of replicator sizes and by the relative time scales and temporal distribution of lifetimes for different dominant species. Yet from this plot alone we have no way to deduce the genealogical paths through which new species emerge; hence we are left with an incomplete picture of the genealogical structure that is itself an integral part of the exploration process.

The graph-based visualization of Figure 6, in contrast, visually expresses the roles played by different species in terms of this very structure. Although the specifics of the visual mapping and graph drawing schemes used here (and described in the previous section) are tentative and constitute only our first attempt at this task, it is clear that the topology revealed in this figure is central to understanding the large-scale dynamics observed in Figure 5. The potential well of small-size species, expressed visually as a rapid cascade to a genealogically segregated, dense, static cluster made up of species *b*, *c*, and *d* in Figure 3, is no longer present in Figure 6. Rather, loop species continue to collectively explore a broad portion of genealogical graph space, roaming to both smaller and larger loops.

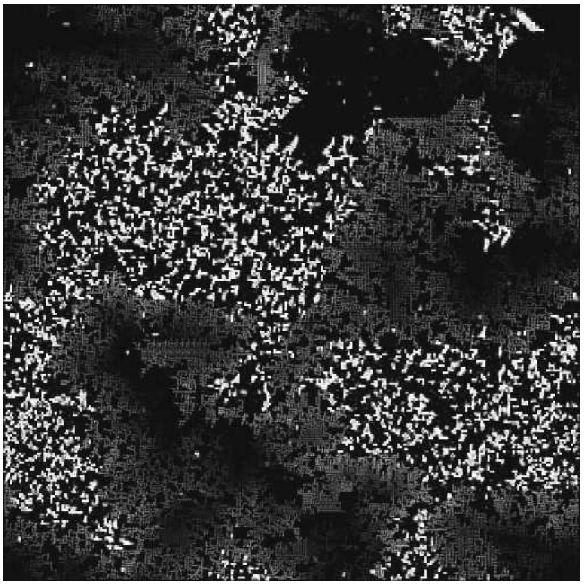

Figure 4. An example of dynamic partitioning of the domain due to the spread of the pathogen over loop populations. The space used is a 3000 × 3000 grid with periodic boundary conditions. Living loops are shown in white, while pathogen cells are shaded according to their remaining lifetime.

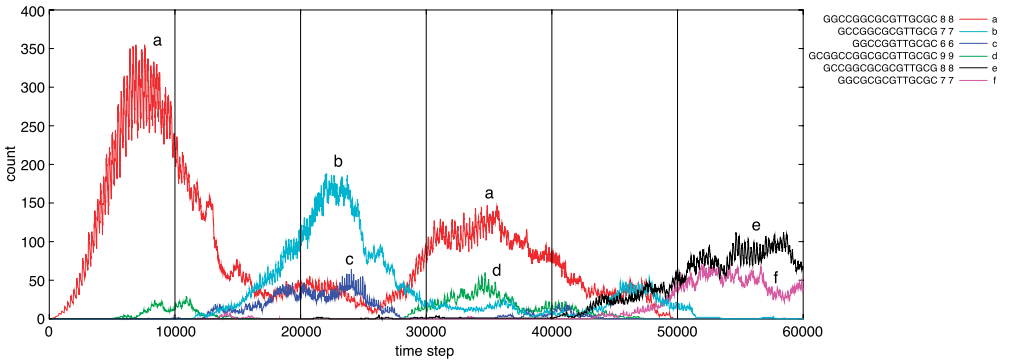

Figure 5. Population dynamics of the Evoloop system with pathogenic environment, i.e., a small block of pathogen cells is introduced into the initial configuration. Counts are plotted only for species whose population exceeds 50 at any time during the simulation run. Dashed vertical lines show divisions between time windows used in Figure 6.

The fact that this graph space exploration occurs can be understood in terms of the partitioning of graph space. From a comparison between Figures 3 and 6 we observe that the number of high-frequency links is significantly reduced due to the local extinction caused by the propagation of the pathogen. In so doing, the addition of a pathogenic environment has enabled the Evoloop system to explore a much more diverse subset of graph space. While this behavior was previously observed [13] via direct data analysis, the visualization illustrated in Figure 6 offers a far more intuitive and understandable representation.

As demonstrated in the above example, the proposed visualization approach using graph-based genealogy successfully captures not only the emergence of new species but also the process leading to their emergence. It also visually shows the critical role played by dynamic habitat partitioning in the maintenance of evolutionary exploration behavior.

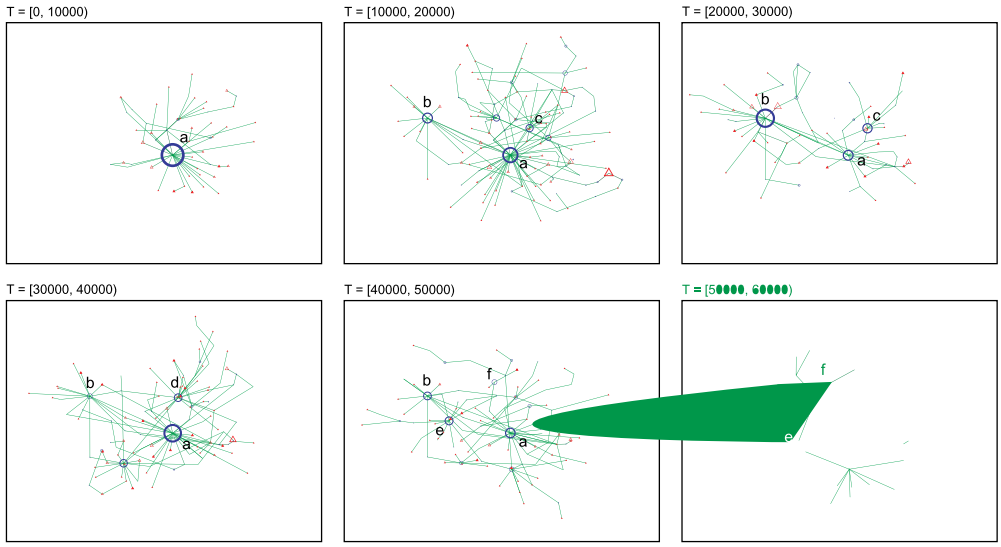

Figure 6. Visualization, using graph-based genealogy analysis, of the evolutionary dynamics with pathogenic environment shown in Figure 5. The sequence of images reads rowwise from left to right, with each image covering 10,000 iterations from T = [0, 10000) to T = [50000, 60000). The x and y coordinates of each node were calculated from the relaxation scheme described in the Appendix.

4 Conclusion

The graph-based approach outlined in this article attempts a new step: to analyze genealogical connectivity by capturing and visualizing the topology of evolutionary transition space of self-replicators in its entirety. Thus, it presents an alternative to standard frequency- or activity-based visualization techniques, and also has a significant benefit in being a general approach capable of representing any arbitrary shape of ancestral connectivity, including evolutionary cycles that would be overlooked in traditional tree-based analyses. Results obtained with the Evoloop example have illustrated the potential of the proposed approach to characterize evolutionary dynamics as flows in genealogical graph space. We believe that this approach provides a better way to analyze and visualize the dynamics of evolutionary systems, especially those where the genealogical structure of replicators is strongly graphlike due to the attraction of populations by natural selection toward a limited state-space regime, typically formulated in quasi-species models [4] and often found in artificial evolutionary models, including the Evoloop. These targets occupy a marginal position between systems described by evolutionary principles and systems described by chemical reaction networks. The dynamics of such marginal systems are of particular interest in recent artificial life and artificial chemistry studies [3], to which our approach presented in this article would be relevant and useful.

As a final caution, we note that the generality we have maintained in this approach has to be paid for in implementation: a problem-specific visualization scheme and—in the more general version—a boundary definition grouping species in identifier space are required. As such, it should be emphasized that we clearly have not introduced an "out-of-the-box" technique. Moreover, it bears repeating that two major assumptions were introduced in the preceding formalism: asexual reproduction (via binary fission), and the availability of a complete data set of all births and deaths, generally appropriate for artificial evolutionary systems yet overly restrictive for the majority of real biological applications. The exemplar case presented in the previous section demonstrates the application of this approach to the analysis of one such artificial system.

This work has been conducted as part of our Artis project, which aims to recreate and analyze self-organized emergent evolutionary behaviors seen in the dynamics of simple, scalable self-replication models in a large-scale domain. For more information on this project, see http://artis.phenome.org.

Acknowledgments

We thank two anonymous reviewers for very helpful and insightful comments in revising this article. H.S. acknowledges partial financial support by a grant from the Hayao Nakayama Foundation for Science, Technology and Culture. C.S. acknowledges additional financial support by grants from the International Information Science Foundation, the Netherlands Organization for International Cooperation in Higher Education (Nuffic), and the VSB Funds.

References

1. Bedau, M., & Brown, C. (1999). Visualizing evolutionary activity of genotypes. *Artificial Life*, *5*, 17–35.
2. Dennett, D. (2002). The new replicators. In M. Pagels (Ed.), *Encyclopedia of evolution* (pp. E83–E92). New York: Oxford University Press.
3. Dittrich, P., Ziegler, J., & Banzhaf, W. (2001). Artificial chemistries—A review. *Artificial Life*, *7*, 225–275.
4. Eigen, M., McCaskill, J., & Schuster, P. (1988). Molecular quasi-species. *Journal of Physical Chemistry*, *92*, 6881–6891.
5. Ikegami, T. (1999). Evolvability of machines and tapes. *Artificial Life and Robotics*, *3*, 242–245.
6. Langton, C. (1984). Self-reproduction in cellular automata. *Physica D*, *10*, 135–144.

7. Lenski, R., Ofria, C., Pennock, R., & Adami, C. (2003). The evolutionary origin of complex features. *Nature*, *423*, 139–144.

8. Pargellis, A. (1996). The evolution of self-replicating computer organisms. *Physica D*, *98*, 111–127.

9. Ray, T. (1991). An approach to the synthesis of life. In C. G. Langton et al. (Eds.), *Artificial Life II* (pp. 371–408). Redwood City, CA: Addison-Wesley.

10. Salzberg, C. (2003). *Emergent evolutionary dynamics of self-reproducing cellular automata*. Master's thesis, Universiteit van Amsterdam, The Netherlands.

11. Salzberg, C., Antony, A., & Sayama, H. (2003). Visualizing evolutionary dynamics of self-replicators using graph-based genealogy. In W. Banzhaf, T. Christaller, P. Dittrich, J. T. Kim, & J. Ziegler (Eds.), *Advances in Artificial Life: Proceedings of the Seventh European Conference on Artificial Life (ECAL2003)* (pp. 387–394). Berlin: Springer-Verlag.

12. Salzberg, C., Antony, A., & Sayama, H. (2004). Complex genetic evolution of self-replicating loops. In J. Pollâck, M. Bedau, P. Husbands, T. Ikegami, & R. A. Watson (Eds.), *Artificial Life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems* (pp. 262–267). Cambridge, MA: MIT Press.

13. Salzberg, C., Antony, A., & Sayama, H. (2004). Evolutionary dynamics of cellular automata-based self-replicators in hostile environments. *BioSystems*, *78*, 119–134.

14. Salzberg, C., & Sayama, H. (2004). Complex genetic evolution of artificial self-replicators in cellular automata. *Complexity*, *10*(2), 33–39.

15. Sayama, H. (1999). A new structurally dissolvable self-reproducing loop evolving in a simple cellular automata space. *Artificial Life*, *5*, 343–365.

16. Schuster, P., & Fontana, W. (1999). Chance and necessity in evolution: Lessons from RNA. *Physica D*, *133*, 427–452.

17. Sipper, M. (1998). Fifty years of research on self-replication: An overview. *Artificial Life*, *4*, 237–257.

18. Szathmáry, E. (1995). A classification of replicators and lambda-calculus models of biological organization. *Proceedings: Biological Sciences*, *260*, 279–286.

19. Yedid, G., & Bell, G. (2002). Macroevolution simulated with autonomously replicating computer programs. *Nature*, *420*, 810–812.

Appendix: Details of the Iterative Relaxation Scheme Used for Graph Visualization

Nodes corresponding to species are initially assigned random real-valued positions $\vec{p}_k$ over the *x*-*y* plane in the range [0, 100] for each axis. Positions are then iteratively updated according to the following relaxation scheme, applied to each node at each update step:

Attractive potential: Each outgoing and incoming link to a given node k is assigned a weight equal to the number of link traversals over the focal time period $\mathbf{T}$. These links are then converted to outgoing vectors from this node according to the relative positions of the target nodes in the visualization plane, and their weighted sum is used to determine the change in position, $\Delta\vec{p}^{\,a}_k$, of node k by attractive forces. Specifically,

$$\Delta\vec{p}^{\,a}_k = \alpha \sum_l (\mathrm{F}(k, l, \mathbf{T}) + \mathrm{F}(l, k, \mathbf{T}))(\vec{p}_l - \vec{p}_k),$$

where α is a rescaling coefficient used to normalize the changes to a given maximum step size.

Repelling potential: A circular neighborhood of given radius is assumed in the visualization plane around node k. The relative positions of all other nodes within this radius are converted to incoming

vectors to this node. The weighted sum of these vectors is used to determine the change in position $\Delta\vec{p}^{\,r}_k$ of node k by repulsive forces. Specifically,

$$\Delta\vec{p}^{\,r}_k = \beta_k \sum_{l\,:\,|\vec{p}_k - \vec{p}_l| < R} w_l(\vec{p}_k - \vec{p}_l),$$

where β_k is another rescaling coefficient that correlates positively with the number of nodes within the neighborhood (crowdedness) and negatively with the sum total of traversal frequencies of incoming and outgoing links on node k (inertia), R is the radius of the assumed repulsion neighborhood, and w_l is the number of links on the target node l. These choices are implemented to avoid local congestion of links in the visualization space and to promote convergence and stabilization of graph structure.

The final update for node k is calculated as $\Delta\vec{p}_k = \Delta\vec{p}^{\,a}_k + \Delta\vec{p}^{\,r}_k$. This iterative procedure is performed for a fixed number of updates, which is set long enough to roughly stabilize the relative positions of nodes.