DECOMPOSITION AND SIMULATION OF SEQUENTIAL DYNAMICAL SYSTEMS

REINHARD LAUBENBACHER AND BODO PAREIGIS

Abstract. Sequential dynamical systems have been developed as a basis for a theory of computer simulation. This paper contains a generalization of this concept. The notion of morphism of sequential dynamical systems is introduced, formalizing the concept of simulating one system by another. Several examples of morphisms are given. Using the morphism concept, it is shown that every sequential dynamical system decomposes uniquely into a product of indecomposable systems.

Introduction

Computer simulation has become an important tool in the study of complex natural and human-made systems, from the biochemical network underlying cell metabolism to road traffic systems in our cities. A variety of simulation tools are available, ranging from discrete event simulations and differential-equations-based simulations, to stochastic simulations, and various hybrids of these.

Much insight has been gained into the structure as well as the dynamic behavior of complex systems through the use of simulations, and they provide an important basis for hypothesis generation, which may determine experimental setups. But simulation is by and large still an art form, with little theoretical guidance to its design, and mostly ad hoc methods for the analysis of the resulting output. Comparison of different simulations of the same system is difficult, as is the comparison of implementations of the same simulation on different platforms, especially of large-scale simulations. Many of the problems associated with simulation approaches require powerful scientific tools, however, as well as a rigorous methodology.

There exists a scattered collection of results and techniques that can be considered part of a newly emerging *simulation science*. An important contribution is the theory of *sequential dynamical systems* (SDS), a mathematical abstraction of a large class of computer simulations [2, 3, 4]. SDS theory is intended as a mathematical foundation for computer simulations that are representable as discrete dynamical systems. Theorems about SDS provide analysis tools for existing simulations and guidance for the design of new ones. Simulation practice has provided the inspiration for the definition of SDS and for the search for theorems about them. A fully developed SDS theory promises to provide answers to many theoretical questions

Date: June 20, 2002.

2000 *Mathematics Subject Classification.* Primary: 37B99, 68Q65, 93A30; Secondary: 18B20, 37B19, 68R01.

This paper was written while the second author was visiting the Physical Science Laboratory of New Mexico State University (NMSU), Las Cruces, in 2001. He wishes to thank the Laboratory and all colleagues for their support and hospitality. The work of the first author was partially supported by funds from a partnership initiative between NMSU and Los Alamos National Laboratory.

about simulations. To the extent that the theory has been used in practice this promise has been fulfilled. In developing simulation science it is important, however, that the theory be guided by a close connection to and intimate knowledge of simulation practice, just as good simulations require intimate knowledge of the systems to be simulated.

Most systems of interest, whether biological, social, or technical, are too large to be simulated accurately. Even if such simulations can be run on a computer, the output is very difficult to analyze. Thus, the question arises how to replace a large system, or large simulation, by a smaller one, and how to relate their dynamics and properties. Such a replacement should come in the form of a "mapping" of some sort between SDS, which carries structural information. We will provide a theoretical framework for this question by defining the notion of a *morphism* of SDS. Morphisms will be closed under composition, and lead therefore to a category of SDS. Based on this framework we will investigate the notion of a morphism in this SDS category as a suitable tool for the simulation of one SDS by another. If such morphisms are indeed the correct way of relating SDS simulations, then a variety of modifications of simulations can be expressed as mathematical constructions, an important step toward the development of mathematical principles for simulation design.

As to the mathematical side of this theory we want to study the dynamical behavior of certain discrete dynamical systems encoded in the structure of the state space of the associated global update function of an SDS. The introduction of permissible maps or morphisms between SDS allows us to observe the effect of structural changes of an SDS on its dynamical behavior. The concept of morphism and category of SDS and other tools used in different categories will help us to eventually develop a mathematical structure theory of SDS and be able to fine tune the dynamic behavior of an SDS. Our approach uses tools from and brings new results to graph theory, discrete mathematics, and discrete dynamical systems theory. The categorical tools used in our approach help us to ask the right questions to understand the relationship between structure and behavior of SDS.

In this paper we describe a more general concept of SDS, and will call the "classical" concept a *permutation* SDS (PSDS). For the convenience of the reader we recall the PSDS concept.

Let $k = \{0, 1\}$. A *permutation sequential dynamical system* $\mathcal{F}$ on the set k^n of binary strings of length n can be thought of as a function

$$f : k^n \longrightarrow k^n,$$

constructed from the following data:

(1) a finite graph F on n vertices,
(2) a family of "local" update functions $f_a : k^n \longrightarrow k^n$, one for each vertex a of F, which changes only the coordinate corresponding to a, and computes the binary state of vertex a. They are furthermore assumed to be symmetric in their inputs. These functions are local in the sense that they only depend on those variables which are connected to a in F.
(3) an "update schedule" π, which specifies an order on the vertices of F, represented by a permutation $\pi \in S_n$.

The function f is then constructed by composing the local functions according to the update schedule π, that is,

$$f = f_{\pi(n)} \circ \cdots \circ f_{\pi(1)} : k^n \longrightarrow k^n.$$

The study of these systems leads to very interesting mathematical questions, independent of applications, and motivated [6], in which we began the development of a more general framework for PSDS. This framework is used in [5] to explore a setup for SDS in which the graph F is not explicit in the data. It naturally suggests a definition for the linearization of a finite system, such as certain types of SDS.

In this paper we generalize the notion of a permutation dynamical system to that of an SDS. To begin with, we allow the set k of states to be arbitrary, rather than just $\{0, 1\}$. In particular, k could be a subset of $\mathbb{R}$, e.g. the interval $[0, 1]$. This could lead to the notions of *fuzzy* and *stochastic* SDS. Secondly, we make no restrictions on the local functions f_a, with respect to symmetry. Most importantly, we use more general update schedules which allow the use of only a subset of all local functions in the construction of the global update function, as well as arbitrary repetitions of local update functions. This is very useful from the point of view of applications. For instance, SDS include global functions that simply permute the entries of a binary string. Such a function cannot be the global update function of a PSDS. This notion of SDS includes in particular PSDS.

The bulk of the paper is devoted to the study of special simulations of SDS by other SDS given by certain reasonable maps between these systems which will lead to the construction of morphisms of SDS, thereby eventually constructing a category **SDS**. From the point of view of applications a morphism from an SDS $\mathcal{F}$ to another SDS $\mathcal{G}$ should be thought of as a simulation of one system by the other. Two interpretations are of particular interest. A monomorphism should represent a simulation of $\mathcal{G}$ by $\mathcal{F}$, then $\mathcal{F}$ is the smaller and more manageable system, $\mathcal{F}$ is a subsystem of $\mathcal{G}$. An epimorphism should represent a simulation of $\mathcal{F}$ by $\mathcal{G}$. Here $\mathcal{G}$ is the smaller system, $\mathcal{G}$ is a quotient system of $\mathcal{F}$.

In our definition of morphisms we are guided by the desire to have sufficiently many morphisms available so that the category would possess finite products, and so that there should be sufficiently many isomorphisms to identify systems that should be considered isomorphic. To determine if two SDS $\mathcal{F}$ and $\mathcal{G}$ are isomorphic is not trivial in view of the very simple Example 2.5 (2). But if we know that $\mathcal{F}$ and $\mathcal{G}$ are isomorphic, then we can pick the easier looking model to study its dynamic behavior.

Furthermore morphisms should help us to find simpler models of SDS as explained below. A morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ of sequential dynamical systems should have the following properties in special cases. $\mathcal{F}$ should mimic to a certain extent the dynamical structure of $\mathcal{G}$, but it should be simpler. For a given sequential dynamical system $\mathcal{G}$ we look for a simple sequential dynamical system $\mathcal{F}$ and a morphism φ, that maps a certain state of $\mathcal{F}$ into a start state of $\mathcal{G}$, so that $\mathcal{F}$ has the "same" dynamical behavior as $\mathcal{G}$ starting at the given state. It would be nice if we could even give a freely generated sequential dynamical system $\mathcal{F}$ with this property. But that seems to be too complicated at present.

One could also consider the dual situation: for a given sequential dynamical system $\mathcal{F}$ together with a start state find a simple sequential dynamical system $\mathcal{G}$ that has the "same" dynamical behavior as $\mathcal{F}$ and find a morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ that maps the start state of $\mathcal{F}$ into $\mathcal{G}$ and that preserves the dynamical behavior.

Thus we want that a morphism of SDS should induce a morphism between their state spaces (phase spaces). That is, there should be a functor from this category to the category of directed graphs. This setup will then help to identify functorially dependent invariants of

SDS, such as its associated global update function or its state space derived from the update function. Certain things may not be invariants of SDS such as the occurrence of certain states e.g. "diagonal states" $(x, \ldots, x)$ in limit cycles of its state space. The result is a rather complex definition of morphism, and we provide a list of examples motivating each of the ingredients.

This categorical setting allows a rigorous study of relations between SDS (and, in particular, of PSDS), in the form of morphisms between them. We show that **SDS** has products, and that every SDS can be decomposed uniquely into a finite product of indecomposable SDS.

1. Sequential Dynamical Systems

We first recall a few facts about graphs and permutation sequential dynamical systems.

Let X be a set and let $\mathcal{P}(X)$ be its power set. Let $\mathcal{P}_2(X) \subseteq \mathcal{P}(X)$ be the subset of all two-element subsets of X.

Definition 1.1. A *(loop free, undirected, finite) graph* $G = (V_G, E_G)$ consists of a finite set V_G of *vertices* and a subset $E_G \subseteq \mathcal{P}_2(X)$ of *edges.*

Definition 1.2. Let F and G be graphs. A *graph morphism* $\varphi : F \longrightarrow G$ consists of a map $\varphi : V_F \longrightarrow V_G$ such that

$$\forall \{a, b\} \in E_F : \{\varphi(a), \varphi(b)\} \in E_G \text{ or } \varphi(a) = \varphi(b),$$

i.e. edges are either mapped to edges or they are 'collapsed' to a vertex.

A subgraph F of a graph G consists of subsets $V_F \subseteq V_G$ and $E_F \subseteq E_G$. The image of a graph morphism is a subgraph. A disjoint union of graphs is a graph. Every graph G can be decomposed into a disjoint union of connected components $G_{(i)}$. A connected subgraph of a graph is always contained in exactly one connected component of the graph. Every graph morphism maps connected components into connected subgraphs.

Let G be a graph. A *1-neighborhood* $N(a)$ of a vertex $a \in V_G$ is the set

$$N(a) := \{b \in V_G \big| \{a, b\} \in E_G \text{ or } a = b\}.$$

Throughout the paper we fix a subcategory Z of the category of sets. Let $V_G = \{a_1, \ldots, a_n\}$. Let $(k[a] | a \in V_G)$ be a family of sets (objects or "zets") in Z. The set $k[a]$ will be called the set of *local states at a.* Define

$$k^n := k[a_1] \times \ldots \times k[a_n] = \prod_{a \in V_G} k[a],$$

the *set of (global) states* of G. We use the following notation. For a state $x \in k^n$ and a vertex $a \in V_G$ we write $x[a]$ for the state of the vertex a or the a-th component of x so that

$$x = (x[a_1], \ldots, x[a_n]).$$

In case that all $k[a]$ are equal to a set k, this definition reduces to the usual definition of k^n.

A function $f : k^n \longrightarrow k^n$ is called *local at $a_i \in V_G$* if

$$f(x[a_1], \ldots, x[a_n]) = (x[a_1], \ldots, x[a_{i-1}], f^i(x[a_1], \ldots, x[a_n]), x[a_{i+1}], \ldots, x[a_n]),$$

where $f^i(x[a_1], \ldots, x[a_n]) \in k[a_i]$ depends only on the variables in the 1-neighborhood $N(a_i)$ of the vertex a_i.

Definition 1.3. A *permutation sequential dynamical system* over the set of states $k = \{0, 1\}$, or a *PSDS* $\mathcal{F} = \big(F, (f_a), \alpha\big)$ consists of

(1) a finite graph F with n vertices,
(2) a family of local functions $(f_a : k^n \longrightarrow k^n \big| a \in V_F, f_a$ local at $a)$, that are symmetric in the arguments,
(3) and a permutation $\alpha = (\alpha(1), \dots, \alpha(n)) = (\alpha_1, \dots, \alpha_n) \in S_n$ of the set V_F of vertices of F, called *update schedule*.

The *global update function* of a PSDS is the function

$$f = f_{\alpha_n} \circ \dots \circ f_{\alpha_1} : k^n \longrightarrow k^n.$$

Definition 1.4. A *sequential dynamical system* or an *SDS*[1], $\mathcal{F} = \big(F, (k[a]), (f_a), \alpha\big)$ consists of

(1) a finite graph F with n vertices,
(2) a family of sets $(k[a] | a \in V_G)$ in Z,
(3) a family of local functions $(f_a : k^n \longrightarrow k^n \big| a \in V_F, f_a$ local at $a)$,
(4) and a word $\alpha = \alpha_{\mathcal{F}} = (\alpha_1, \dots, \alpha_r) \in V_F^*$ in the Kleene closure of the set of vertices V_G, called *update schedule* (i.e. a map $\alpha : \{1, \dots, r\} \longrightarrow V_F$).

The word α is used to define the *global update function* of an SDS as the function

$$f = f_{\alpha_r} \circ \dots \circ f_{\alpha_1} : k^n \longrightarrow k^n.$$

The *length* of the update schedule $\alpha = (\alpha_1, \dots, \alpha_r)$ is r. The global update function of an SDS defines its dynamical behavior, properties of limit cycles, transients, etc..

If we choose $k[a] = \{0, 1\}$, α a permutation in S_n, and assume that the f_a are symmetric in the arguments, then we obtain the definition of a permutation sequential dynamical system.

2. Morphisms of Sequential Dynamical Systems

In order to define a morphism of SDS we have to consider the following data of an SDS: the graph F, the local functions (f_a), the word α, and the sets of states $(k[a])$, all of which may be changed by maps. For the graph F, the word α, and the sets of states $k[a]$ we shall introduce maps with certain compatibility requirements. The local functions f_a will occur in commutative diagrams.

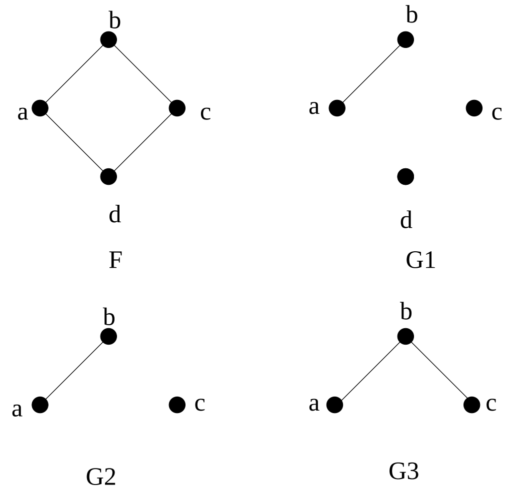
Definition 2.1. Let $\mathcal{F} = \big(F, (k[a]), (f_i : k^n \longrightarrow k^n), \alpha\big)$ (with $V_F = \{a_1, \dots, a_n\}$ and $f_i = f_{a_i}$) and $\mathcal{G} = \big(G, (k[b]), (g_j : k^m \longrightarrow k^m), \beta\big)$ be SDS. Let $\varphi_g : G \longrightarrow F$ be a graph morphism, and $(\varphi_s[b] : k[\varphi_g(b)] \longrightarrow k[b] | b \in V_G)$ be a family of maps in the category Z. Then φ_g and the family $(\varphi_s[b])$ induce an adjoint map on the state spaces as follows: consider the pairing

$$k^n \times V_F \ni (x, a) \mapsto \langle x, a\rangle := x[a] \in \bigcup_{a \in V_F} k[a],$$

and similarly $k^m \times V_G \longrightarrow \bigcup k[b]$. Then $\varphi_g : G \longrightarrow F$ and $(\varphi_s[b])$ induce an *adjoint map* $\varphi^* : k^n \longrightarrow k^m$ with

$$\langle \varphi^*(x), b\rangle := \varphi_s[b](\langle x, \varphi_g(b)\rangle) \tag{1}$$

[1]Subsequently, we will use the acronym SDS for plural as well as singular instances.



or

$$\varphi^*(x[a_1], \ldots, x[a_n]) := (\varphi_s[b_1](x[\varphi_g(b_1)]), \ldots, \varphi_s[b_m](x[\varphi_g(b_m)])).$$

Remark 2.2. Let $(G, (k[b]), (g_j : k^m \longrightarrow k^m), \beta)$ be an SDS. Let $\{G_{(l)}\}$ be the set of connected components of G. Let $g_i : k^m \longrightarrow k^m$ and $g_j : k^m \longrightarrow k^m$ be two local functions for the vertices a_i, a_j in different connected components, then $g_i \circ g_j = g_j \circ g_i$, since both maps depend only on the 1-neighborhoods of a_i and a_j contained in the disjoint connected components. Similarly any two products of local functions $f_{i_1} \circ \ldots \circ f_{i_r}$ all being defined over a connected component $G_{(1)}$ and $f_{j_1} \circ \ldots \circ f_{j_s}$ all being defined over a connected component $G_{(2)}$ commute.

Let $\mathcal{F} = \big(F, (k[a]), (f_i : k^n \longrightarrow k^n), \alpha\big)$ and $\mathcal{G} = \big(G, (k[b]), (g_j : k^m \longrightarrow k^m), \beta\big)$ be SDS. As first components of a morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ we have already a map of graphs $\varphi_g : G \longrightarrow F$ and a family of maps $(\varphi_s[b] : k[\varphi_g(b)] \longrightarrow k[b] | b \in V_G)$ in Z (together with its adjoint map $\varphi^* : k^n \longrightarrow k^m$). With the following examples we want to motivate the conditions that we have to impose on maps between words that relate α and β.

We want that morphisms between SDS should preserve the local and global dynamical behavior. This implies that morphisms between SDS lead to morphisms between the associated global update functions. The following diagram should commute:

$$\begin{tikzcd}
k^n \arrow[r, "f"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"] \\
k^m \arrow[r, "g"] & k^m
\end{tikzcd}$$

We use the word $\alpha = (\alpha_1, \ldots, \alpha_s)$ to describe the global update function as $f = f_{\alpha_s} \circ \cdots f_{\alpha_1}$ and $\beta = (\beta_1, \ldots, \beta_t)$ for $g = g_{\beta_t} \circ \cdots g_{\beta_1}$. We want to reduce the above diagram to similar conditions about the local update functions. The followings examples will show the most important points that have to be observed for this definition.

Examples 2.3. In the following list of examples we refer to the graphs in Figure 1. We assume $k[a] = k[b] = k$ for all $a \in V_F$, $b \in V_G$, and $\varphi_s[b] = \mathrm{id}$ for all $b \in V_G$.

(1) Let $\varphi_g : G_1 \longrightarrow F$ be the identity map on the vertices. Let $\mathcal{F}$ be defined with the word $\alpha = (abcd)$, and let $\mathcal{G}$ be defined with the word $\beta = (abcd)$. If we require that the following

diagrams commute

$$\begin{tikzcd} k^n \arrow[r, "f_{\alpha_i}"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"] \\ k^m \arrow[r, "g_{\beta_i}"] & k^m \end{tikzcd}$$

then the diagram

$$\begin{tikzcd} k^n \arrow[r, "f_a"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_b"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_c"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_d"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"'] \\ k^m \arrow[r, "g_a"] & k^m \arrow[r, "g_b"] & k^m \arrow[r, "g_c"] & k^m \arrow[r, "g_d"] & k^m \end{tikzcd}$$

commutes, and thus we have a commutative diagram for the global update functions.

(2) Consider the graph morphism $\varphi_g : G_2 \longrightarrow F$, given by the inclusion on the vertices. We use the words $\alpha = (abcd)$ and $\beta = (abc)$. Then we are forced to require that the following diagram commutes.

$$\begin{tikzcd} k^n \arrow[r, "f_a"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_b"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_c"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_d"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"'] \\ k^m \arrow[r, "g_a"] & k^m \arrow[r, "g_b"] & k^m \arrow[r, "g_c"] & k^m \arrow[r, "\mathrm{id}"] & k^m \end{tikzcd}$$

So we have to assume that the partial diagram concerning f_d commutes with the identity as lower arrow, since $\varphi_g^{-1}(d) = \emptyset$. Again we have that the diagram for the global update functions commutes.

(3) Use $\varphi_g : G_3 \longrightarrow F$, with $\varphi_g(a) = a$ and $\varphi_g(b) = \varphi_g(c) = b$, and $\alpha = (abcd)$, resp. $\beta = (abc)$. Then we have to consider the following diagram:

$$\begin{tikzcd} k^n \arrow[r, "f_a"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_b"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_c"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_d"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"'] \\ k^m \arrow[r, "g_a"] & k^m \arrow[r, "g_c g_b"] & k^m \arrow[r, "\mathrm{id}"] & k^m \arrow[r, "\mathrm{id}"] & k^m \end{tikzcd}$$

Observe that there are no vertices being mapped into $c, d \in V(F)$, but that the two vertices b and c in G_3 are mapped to b in F. Obviously the order of g_b and g_c is important. So a map between the words of the two SDS should not only be compatible with the graph map but it should also be order preserving in some sense.

(4) Now we consider $\varphi_g : G_2 \longrightarrow F$, resp. $\varphi_g : G_3 \longrightarrow F$, sending a to a, b to b, and c to a, and $\alpha = (abcd)$, resp. $\beta = (abc)$. Then we have to consider the following diagram:

$$\begin{tikzcd} k^n \arrow[r, "f_a"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_b"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_c"] \arrow[d, "\varphi^*"'] & k^n \arrow[r, "f_d"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"'] \\ k^m \arrow[r, "g_c g_a"] & k^m \arrow[r, "g_b"] & k^m \arrow[r, "\mathrm{id}"] & k^m \arrow[r, "\mathrm{id}"] & k^m \end{tikzcd}$$

Observe that the composition in the lower row gives $g_b g_c g_a$. In case of the graph G_2 this is the global update function $g = g_c g_b g_a$, since b and c are in different connected components of the graph, so that g_b and g_c commute under composition. In the case of the graph G_3, however, these two local update functions do not commute in general, so that this construction should not give a morphism in our category, and we have to exclude it. This can be done by looking at order preserving maps from the subwords on the connected components of G_2, resp. G_3, to the word α on the graph F.

Now we fix some notation that we will use in the following definition.

Let $\beta_{(l)}$ denote the subword of β whose letters belong to the connected component $G_{(l)}$. Let $|\beta|$ denote the ordered set of indices $\{1, \ldots,$ length of $\beta\}$ and let $|\beta_{(l)}|$ denote the ordered subset of indices of $\beta_{(l)}$ (in $|\beta|$).

Observe that the (unordered) set $|\beta|$ decomposes into a disjoint union of the (unordered) sets $|\beta_{(l)}|$.

Definition 2.4. Let $\mathcal{F} = \big(F, (k[a]), (f_i : k^n \longrightarrow k^n), \alpha\big)$ (with $V_F = \{a_1, \ldots, a_n\}$ and $f_i = f_{a_i}$) and $\mathcal{G} = \big(G, (k[b]), (g_j : k^m \longrightarrow k^m), \beta\big)$ (with $V_G = \{b_1, \ldots, b_m\}$) be SDS.

A *(Z-)morphism of sequential dynamical systems* $\varphi : (F, (k[a]), (f_i : k^n \longrightarrow k^n), \alpha) \longrightarrow (G, (k[b]), (g_j : k^m \longrightarrow k^m), \beta)$ consists of

- a graph morphism $\varphi_g : G \longrightarrow F$ (reverse direction!),
- a family of maps $(\varphi_s[b] : k[\varphi_g(b)] \longrightarrow k[b] \mid b \in V_G, \varphi_s[b] \in Z)$,
- and a family of order preserving maps

$$\widetilde{\varphi}_{(l)} : |\beta_{(l)}| \longrightarrow |\alpha|$$

  for each connected component $G_{(l)}$ of G

such that

- $\forall l \quad \forall j \in |\beta_{(l)}| : \varphi_g(\beta_j) = \alpha_{\tilde{\varphi}_{(l)}(j)}$, i.e. all $\widetilde{\varphi}_{(l)}$ are compatible with the given graph morphism φ_g.
- if $i \in |\alpha|$ and $\overline{\varphi}_{(l)}(i) := (\beta_j | \widetilde{\varphi}_{(l)}(j) = i)$ is the subword of $\beta_{(l)}$ mapped into α_i then the diagram

$$\begin{tikzcd} k^n \arrow[r, "f_{\alpha_i}"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"] \\ k^m \arrow[r, "\prod_l \prod_{\overline{\varphi}_{(l)}(i)} g_{\beta_j}"] & k^m \end{tikzcd}$$

  commutes, where the product $\prod_{\overline{\varphi}_{(l)}(i)} g_{\beta_j}$ is taken in the order of the entries in the subword $\overline{\varphi}_{(l)}(i)$. (If $\overline{\varphi}_{(l)}(i)$ is the empty word, then the product is assumed to be the identity map.)

In order to further motivate this definition we give a few examples that show that the conditions are necessary for our further studies.

Example 2.5. (1) This example shows that two very simple SDS that should be isomorphic, are indeed isomorphic. Here we use $k[a] = k[b] = k$ and we need the set maps $\varphi_s[b] : k \longrightarrow k$.

Let $F = \{a\}$ be the one vertex graph. For $i \in k$ define the set map $p_i : k \longrightarrow k$ to be the projection of k onto the element $i \in k$. Let $i, j \in k$ with $i \neq j$. Let $\mathcal{F} = \big(F, (k), (p_i : k \longrightarrow k), (a)\big)$ and $\mathcal{G} = \big(F, (k), (p_j : k \longrightarrow k), (a)\big)$. Then $\mathcal{F} \cong \mathcal{G}$.

We define the isomorphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ by $\varphi_g = \mathrm{id} : \{a\} \longrightarrow \{a\}$, $\varphi_s[a] = \pi : k \longrightarrow k$ where π is some bijective map from k to k with $\pi(i) = j$. Finally let $\widetilde{\varphi}_{(1)} = \mathrm{id}$. Then φ is a morphism of SDS with inverse $\psi = (\mathrm{id}, \pi^{-1}, \mathrm{id})$. In fact we have $\varphi_g(\beta_1) = \mathrm{id}(a) = \alpha_{\widetilde{\varphi}_{(1)}(1)}$ and the diagram

$$\begin{tikzcd} k \arrow[r, "p_i"] \arrow[d, "\varphi^*=\pi"'] & k \arrow[d, "\varphi^*=\pi"] \\ k \arrow[r, "p_j"] & k \end{tikzcd}$$

commutes.

(2) Consider the following two SDS $\mathcal{F} = \big(F, (k), (f_i), \alpha\big)$ and $\mathcal{G} = \big(F, (k), (g_i), \alpha\big)$ over $k = \{0, 1\}$ where $V_F = \{a, b\}$ and $E_F = \big\{\{a, b\}\big\}$. Let

$$\begin{aligned} f_a(x_1, x_2) &= (x_2, x_2), & f_b(x_1, x_2) &= (x_1, \overline{x_2}), \\ g_a(x_1, x_2) &= (\overline{x_2}, x_2), & g_b(x_1, x_2) &= (x_1, \overline{x_2}). \end{aligned}$$

The update schedule α is arbitrary. It is not clear if these SDS are isomorphic, in particular if they give isomorphic state spaces as described in section 3. If we define $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ by $\varphi_g = \mathrm{id}$, $\varphi_s[a] = \tau$, $\varphi_s[b] = \mathrm{id}$, and $\widetilde{\varphi}_{(1)} = \mathrm{id}$ then this is an isomorphism since the diagrams

$$\begin{tikzcd} k^2 \arrow[r, "f_a"] \arrow[d, "\varphi^*=\tau\times\mathrm{id}"'] & k^2 \arrow[d, "\varphi^*=\tau\times\mathrm{id}"] \\ k^2 \arrow[r, "g_a"] & k^2 \end{tikzcd} \qquad \begin{tikzcd} k^2 \arrow[r, "f_b"] \arrow[d, "\varphi^*=\tau\times\mathrm{id}"'] & k^2 \arrow[d, "\varphi^*=\tau\times\mathrm{id}"] \\ k^2 \arrow[r, "g_b"] & k^2 \end{tikzcd}$$

commute.

(3) We determine the set of all morphisms in some very simple cases. Let $k[a] = k[b] = k = \{0, 1\}$. We consider 4 SDS all defined on the trivial one point graph $\{a\}$ and with the trivial one letter update schedule a. Let

$$\begin{aligned} \mathcal{I} &:= (a, k, \mathrm{id} : k \longrightarrow k, a), \\ \mathcal{T} &:= (a, k, \tau : k \longrightarrow k, a), \\ \mathcal{P}_0 &:= (a, k, p_0 : k \longrightarrow k, a), \\ \mathcal{P}_1 &:= (a, k, p_1 : k \longrightarrow k, a), \end{aligned}$$

where τ is the interchange map sending 1 to 0 and 0 to 1, and the p_i are the maps in the preceding example. In order to find isomorphic SDS, we have to find out if the diagram

$$\begin{tikzcd} k \arrow[r, "f"] \arrow[d, "\varphi^*"'] & k \arrow[d, "\varphi^*"] \\ k \arrow[r, "g"] & k \end{tikzcd}$$

commutes. If $f \neq g$ this is only the case for $f = p_0$ and $g = p_1$ (or conversely). Thus we have that three of the four SDS are nonisomorphic. More generally we have for the number of morphisms

$$\begin{array}{llll} |\mathrm{Mor}(\mathcal{I}, \mathcal{I})| = 4 & |\mathrm{Mor}(\mathcal{T}, \mathcal{T})| = 2 & |\mathrm{Mor}(\mathcal{P}_0, \mathcal{P}_0)| = 2 & |\mathrm{Mor}(\mathcal{P}_1, \mathcal{P}_1)| = 2 \\ |\mathrm{Mor}(\mathcal{I}, \mathcal{T})| = 0 & |\mathrm{Mor}(\mathcal{T}, \mathcal{I})| = 2 & |\mathrm{Mor}(\mathcal{P}_0, \mathcal{P}_1)| = 2 & |\mathrm{Mor}(\mathcal{P}_1, \mathcal{P}_0)| = 2 \\ |\mathrm{Mor}(\mathcal{I}, \mathcal{P}_0)| = 1 & |\mathrm{Mor}(\mathcal{P}_0, \mathcal{I})| = 2 & |\mathrm{Mor}(\mathcal{T}, \mathcal{P}_0)| = 1 & |\mathrm{Mor}(\mathcal{P}_0, \mathcal{T})| = 0 \\ |\mathrm{Mor}(\mathcal{I}, \mathcal{P}_1)| = 1 & |\mathrm{Mor}(\mathcal{P}_1, \mathcal{I})| = 2 & |\mathrm{Mor}(\mathcal{T}, \mathcal{P}_1)| = 1 & |\mathrm{Mor}(\mathcal{P}_1, \mathcal{T})| = 0. \end{array}$$

(4) We consider now some morphisms that only depend on the choice of the order preserving maps $\widetilde{\varphi}_{(l)}$ but that are defined on the same morphism φ_g of graphs.

Let F be defined by $V_F = \{a_1, a_2\} = \{u, v\}$ (with $u = a_1$, $v = a_2$) and $E_F = \{\{u, v\}\}$, and let $\alpha = (u, v, u)$. Let $k[u] = k[v] = k$. At this time we do not fix the local functions $f_u, f_v : k^2 \longrightarrow k^2$. Furthermore let G be defined by $V_G = \{b_1\} = \{w\}$ and $E_G = \emptyset$, let $k[w] = k$, and let $\beta = (w, w)$. For G we also do not fix the local function $g_w : k \longrightarrow k$. Define a graph morphism $\varphi_g : G \longrightarrow F$ by $\varphi_g(w) = u$. We also fix $\varphi_s[w] : k \longrightarrow k$ to be the identity. Then $\varphi^* : k^2 \longrightarrow k$ is the projection onto the first component pr_1, since $\varphi^*(x[u], x[v]) = x[\varphi_g(w)] = x[u]$. Both graphs have only one connected component. Our definitions give ordered sets $|\alpha| = \{1, 2, 3\}$, $|\beta| = \{1, 2\} = |\beta_{(1)}|$.

For $\widetilde{\varphi}_{(l)} : |\beta_{(1)}| \longrightarrow |\alpha|$ or $\widetilde{\varphi}_{(l)} : \{1, 2\} \longrightarrow \{1, 2, 3\}$ we have the choice among three "order preserving maps of ordered multisets" that satisfy the first axiom for morphisms:

$$\begin{tikzcd}[column sep=small] w \arrow[d] & w \arrow[dl] & \\ u & v & u \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] w \arrow[d] & w \arrow[dr] & \\ u & v & u \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] w \arrow[drr] & w \arrow[dr] & \\ u & v & u \end{tikzcd}$$

We want to establish conditions so that these three maps define morphisms of SDS.

Case 1: We have $\widetilde{\varphi}_{(1)}(1) = 1$ and $\widetilde{\varphi}_{(1)}(2) = 1$. So the following diagrams must commute:

$$\begin{tikzcd} k^2 \arrow[r, "f_{\alpha_1} = f_u"] \arrow[d, "\mathrm{pr}_1"'] & k^2 \arrow[r, "f_{\alpha_2} = f_v"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[r, "f_{\alpha_3} = f_u"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[d, "\mathrm{pr}_1"] \\ k \arrow[r, "g_{\beta_2} g_{\beta_1} = g_w^2"'] & k \arrow[r, "\mathrm{id}"'] & k \arrow[r, "\mathrm{id}"'] & k \end{tikzcd}$$

Conditions for the local maps so that this becomes a morphism of SDS are $f_u = \mathrm{id}$ and $g_w^2 = \mathrm{id}$.

Case 2: We have $\widetilde{\varphi}_{(1)}(1) = 1$ and $\widetilde{\varphi}_{(1)}(2) = 3$. So the following diagrams must commute:

$$\begin{tikzcd} k^2 \arrow[r, "f_{\alpha_1} = f_u"] \arrow[d, "\mathrm{pr}_1"'] & k^2 \arrow[r, "f_{\alpha_2} = f_v"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[r, "f_{\alpha_3} = f_u"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[d, "\mathrm{pr}_1"] \\ k \arrow[r, "g_{\beta_1} = g_w"'] & k \arrow[r, "\mathrm{id}"'] & k \arrow[r, "g_{\beta_1} = g_w"'] & k \end{tikzcd}$$

Conditions for the local maps so that this becomes a morphism of SDS are $f_u(x[u], x[v]) = (g_w(x[u]), x[v])$.

Case 3: We have $\widetilde{\varphi}_{(1)}(1) = 3$ and $\widetilde{\varphi}_{(1)}(2) = 3$. So the following diagrams must commute:

$$\begin{tikzcd} k^2 \arrow[r, "f_{\alpha_1} = f_u"] \arrow[d, "\mathrm{pr}_1"'] & k^2 \arrow[r, "f_{\alpha_2} = f_v"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[r, "f_{\alpha_3} = f_u"] \arrow[d, "\mathrm{pr}_1"] & k^2 \arrow[d, "\mathrm{pr}_1"] \\ k \arrow[r, "\mathrm{id}"'] & k \arrow[r, "\mathrm{id}"'] & k \arrow[r, "g_{\beta_2} g_{\beta_1} = g_w^2"'] & k \end{tikzcd}$$

Conditions for the local maps so that this becomes a morphism of SDS are $f_u = \mathrm{id}$ and $g_w^2 = \mathrm{id}$.

(5) If $k[a] = [0, 1] \subset \mathbb{R}$ is the closed unit interval and $k[b] = \{0, \frac{1}{2}, 1\}$, then any SDS with state spaces $k[a]$ can be considered as a probabilistic or fuzzy dynamical system, where each

vertex has states between 0 and 1. Some interesting local update functions $f_a : k^n \longrightarrow k^n$ are the ones where the a-th component is the product of all states in a 1-neighborhood of x_a. A discretization of such a system is obtained by taking the identity maps for φ_g and $\widetilde{\varphi}$, and as maps between the states the maps that send 0 to 0, 1 to 1, and all other values to $\frac{1}{2}$.

Theorem 2.6. *The sequential dynamical systems $(F, (k[a]), (f_a), \alpha)$ together with the morphisms of SDS form a category* **SDS**.

Proof. The associativity and unit laws are easily checked. So the proof follows from $\square$

Proposition and Definition 2.7. *Let*

$$\varphi : \mathcal{F} = \big(F, (k[a]), (f_1, \dots, f_n), \alpha\big) \longrightarrow \mathcal{G} = \big(G, (k[b]), (g_1, \dots, g_m), \beta\big)$$

and

$$\psi : \mathcal{G} = \big(G, (k[b]), (g_1, \dots, g_m), \beta\big) \longrightarrow \mathcal{H} = \big(H, (k[c]), (h_1, \dots, h_r), \gamma\big)$$

be two morphisms of SDS. Then the composition

$$\psi \circ \varphi : \mathcal{F} \longrightarrow \mathcal{H}$$

is a morphism of SDS, where the composition $\psi \circ \varphi$ consists of

- *the composite of the associated graph morphisms $\varphi_g \psi_g : H \longrightarrow F$,*
- *the family of maps*

$$(\psi \circ \varphi)_s[c] := \psi_s[c]\varphi_s[\psi_g(c)] \text{ for all } c \in H, \tag{2}$$

- *and the composite of the corresponding families of order preserving maps $\widetilde{\psi}_{(l)} : |\gamma_{(l)}| \longrightarrow |\beta|$ and $\widetilde{\varphi}_{(l')} : |\beta_{(l')}| \longrightarrow |\alpha|$.*

Proof. The composite of the graph morphisms is obviously again a graph morphism.

Since connected components of the graph H are mapped into connected components of the graph G, we find a connected component $G_{(l')}$ into which the connected component $H_{(l)}$ is mapped hence we get $\widetilde{\psi}_{(l)} : |\gamma_{(l)}| \longrightarrow |\beta_{(l')}|$ so that the order preserving maps can be composed to

$$\widetilde{(\varphi\psi)}_{(l)} : |\gamma_{(l)}| \overset{\widetilde{\psi}_{(l)}}{\longrightarrow} |\beta_{(l')}| \overset{\widetilde{\varphi}_{(l')}}{\longrightarrow} |\alpha|.$$

The first axiom is easily verified: let $H_{(l)}$ be a connected component of H. Let $j \in |\gamma_{(l)}|$. Then $\varphi_g \psi_g(\gamma_j) = \varphi_g(\beta_{\widetilde{\psi}_{(l)}(j)}) = \alpha_{\widetilde{\varphi}_{(l')}\widetilde{\psi}_{(l)}(j)} = \alpha_{\widetilde{(\varphi\psi)}_{(l)}}(j)$.

Now observe that

$$\begin{aligned}
\langle (\psi \circ \varphi)^*(x), c \rangle &= (\psi \circ \varphi)_s[c](\langle x, (\psi \circ \varphi)_g(c) \rangle) \\
&= \psi_s[c]\varphi_s[\psi_g(c)](\langle x, \varphi_g \psi_g(c) \rangle) \\
&= \psi_s[c](\langle \varphi^*(x), \psi_g(c) \rangle) \\
&= \langle \psi^* \varphi^*(x), c \rangle
\end{aligned}$$

which implies $(\psi \circ \varphi)^* = \psi^*\varphi^*$. So we have to show that the diagram

$$\begin{tikzcd}
k^n \arrow[r, "f_{\alpha_i}"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"] \\
k^m \arrow[r, "\prod g_{\beta_j}"] \arrow[d, "\psi^*"'] & k^m \arrow[d, "\psi^*"] \\
k^r \arrow[r, "\prod h_{\gamma_u}"] & k^r
\end{tikzcd}$$

commutes. The middle arrow can be decomposed into $\prod_l \prod_{\overline{\varphi}_{(l)}(i)} g_{\beta_j} = g_{\beta_t} \dots g_{\beta_1}$ and for each g_{β_j} we get associated diagrams from ψ so that the total diagram

$$\begin{tikzcd}
k^n \arrow[rrrrr, "f_{\alpha_i}"] \arrow[d, "\varphi^*"'] & & & & & k^n \arrow[d, "\varphi^*"] \\
k^m \arrow[r, "g_{\beta_1}"] \arrow[d, "\psi^*"'] & k^m \arrow[d, "\psi^*"] & \cdots & k^m \arrow[rr, "g_{\beta_t}"] \arrow[d, "\psi^*"'] & & k^m \arrow[d, "\psi^*"] \\
k^r \arrow[r, "\prod_l \prod_{\overline{\psi}_{(l)}(1)} h_{\gamma_j}"] & k^r & \cdots & k^r \arrow[rr, "\prod_l \prod_{\overline{\psi}_{(l)}(t)} h_{\gamma_j}"] & & k^r
\end{tikzcd}$$

commutes. From the considerations we made about the composition of $\widetilde{\psi}_{(l)}$ and $\widetilde{\varphi}_{(l')}$ it follows that the product on the lower line is $\prod_l \prod_{\overline{\varphi\psi}_{(l)}(i)} h_{\gamma_j}$ by using the fact from Remark 2.2 that local functions on different connected components commute:

$$\prod_l \prod_{\overline{\psi}_{(l)}(t)} h_{\gamma_j} \dots \prod_l \prod_{\overline{\psi}_{(l)}(1)} h_{\gamma_j} = \prod_l \left(\prod_{\overline{\psi}_{(l)}(t) \dots \overline{\psi}_{(l)}(1)} h_{\gamma_j} \right) = \prod_l \left(\prod_{\overline{\varphi\psi}_{(l)}(i)} h_{\gamma_j} \right),$$

where $\overline{\psi}_{(l)}(t) \dots \overline{\psi}_{(l)}(1)$ is the concatenation of the corresponding subwords. This is a subword of γ since $\widetilde{\psi}_{(l)}$ is order preserving and $(\beta_t, \dots, \beta_1)$ is a subword of β.

Hence the diagrams

$$\begin{tikzcd}
k^n \arrow[r, "f_{\alpha_i}"] \arrow[d, "(\psi\circ\varphi)^*"'] & k^n \arrow[d, "(\psi\circ\varphi)^*"] \\
k^r \arrow[r, "\prod_l \prod_{\overline{\varphi\psi}_{(l)}(i)} h_{\gamma_j}"] & k^r
\end{tikzcd}$$

commute. □

Since the composition of morphisms of SDS is defined by composition of certain set maps, isomorphisms of SDS consists of certain bijective set maps. This is useful to construct or identify isomorphisms. In fact Example 2.5 (2) was constructed in such a way.

3. State Spaces

Any function $f : k^n \longrightarrow k^n$ defines a finite directed graph with vertex set k^n and directed edges $(x, f(x))$ for all $x \in k^n$, called the *state space* of $f : k^n \longrightarrow k^n$. A morphism from $f : k^n$

$\longrightarrow k^n$ to $g : k^m \longrightarrow k^m$ is a commutative diagram

$$\begin{tikzcd} k^n \arrow[r, "f"] \arrow[d, "h"'] & k^n \arrow[d, "h"] \\ k^m \arrow[r, "g"] & k^m \end{tikzcd} .$$

In this way every morphism of 'functions' induces a morphism of the associated state spaces in the category of directed graphs. So we have a covariant functor from 'functions' to the full subcategory $\mathbf{S}$ of state spaces in the category of directed graphs.

Let $\mathcal{F} = (F, (k[a]), (f_1, \dots, f_n), \alpha)$ be an SDS with update function $f_\alpha : k^n \longrightarrow k^n$, $f_\alpha := f_{\alpha_1} \dots f_{\alpha_n}$. In this section we show that there is a covariant functor

$$\mathcal{S} : \mathbf{SDS} \longrightarrow \mathbf{S},$$

given by assigning to an SDS the state space of its update function.

Lemma 3.1. *A morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ of SDS induces a commutative diagram*

$$\begin{tikzcd} k^n \arrow[r, "f_\alpha"] \arrow[d, "\varphi^*"'] & k^n \arrow[d, "\varphi^*"] \\ k^m \arrow[r, "g_\beta"] & k^m \end{tikzcd} ,$$

that is a morphism of the update functions. Furthermore φ induces a graph morphism $\mathcal{S}(\varphi)$ between the state space of $f_\alpha : k^n \longrightarrow k^n$ and the state space of $g_\beta : k^m \longrightarrow k^m$.

Proof. The diagram

$$\begin{tikzcd} k^n \arrow[r, "f_{\alpha_1}"] \arrow[d, "\varphi^*"'] & k^n \arrow[r] \arrow[d, "\varphi^*"] & \dots \arrow[r] & k^n \arrow[r, "f_{\alpha_r}"] \arrow[d, "\varphi^*"] & k^n \arrow[d, "\varphi^*"] \\ k^m \arrow[r, "\prod_l \prod_{\overline{\varphi}_{(l)}(1)} g_{\beta_j}"] & k^m \arrow[r] & \dots \arrow[r] & k^m \arrow[r, "\prod_l \prod_{\overline{\varphi}_{(l)}(r)} g_{\beta_j}"] & k^m \end{tikzcd}$$

commutes. It remains to show that

$$g_\beta = g_{\beta_s} \dots g_{\beta_1} = \prod_l \prod_{\overline{\varphi}_{(l)}(r)} g_{\beta_j} \cdots \prod_l \prod_{\overline{\varphi}_{(l)}(1)} g_{\beta_j}.$$

Since the g_{β_j} are local functions, the functions g_{β_i} and g_{β_j} commute if their vertices β_i and β_j are in different connected components of G. Hence it suffices to show that

$$\prod_{\beta_j \in \beta_{(l)}} g_{\beta_j} = \prod_{\beta_j \in \overline{\varphi}_{(l)}(r)} g_{\beta_j} \cdots \prod_{\beta_j \in \overline{\varphi}_{(l)}(1)} g_{\beta_j}.$$

In fact the g's may be grouped together according to connected components without changing their product. But this equation holds since the map $\tilde{\varphi}_{(l)} : |\beta_{(l)}| \longrightarrow |\alpha|$ is order preserving. The rest of the statement follows trivially. $\square$

Thus φ induces a morphism of state spaces

$$\mathcal{S}(\varphi) : \mathcal{S}(\mathcal{F}) \longrightarrow \mathcal{S}(\mathcal{G}).$$

This proves the following theorem.

Theorem 3.2. *Passage to state spaces induces a covariant functor*

$$\mathcal{S} : \mathbf{SDS} \to \mathbf{S}.$$

Observe that this functor extracts and clarifies the dynamic behavior of an SDS.

4. Products of Sequential Dynamical Systems

Theorem 4.1. *The category* **SDS** *has finite products.*

Proof. Let $\mathcal{F} = (F, (k[a]), (f_1, \ldots, f_n), \alpha)$ and $\mathcal{G} = (G, (k[b]), (g_1, \ldots, g_m), \beta)$ be SDS. Define the *product*

$$\mathcal{H} = (H, (k[c]), (h_1, \ldots, h_{n+m}), \gamma) = \mathcal{F} \times \mathcal{G}$$

as follows. Let $H = F \dot\cup G$ be the disjoint union of the graphs F and G, with vertex set $V_F \dot\cup V_G$. Observe that the graph components F and G are disconnected. Define $k[c] := k[a]$ for any $c = a \in V_F \subseteq V_H$ and $k[c] := k[b]$ for any $c = b \in V_G \subseteq V_H$. Now define

$$h_c : k^{n+m} = k^n \times k^m \longrightarrow k^{n+m}$$

as follows. For $c = a \in V_F$ define $h_c := f_a \times \mathrm{id}$, and for $c = b \in V_G$ we set $h_c := \mathrm{id} \times g_b$. Finally, we define the update schedule as the concatenation of words

$$\gamma = \alpha\beta.$$

Next we define a projection morphism

$$\mathrm{pr}_{\mathcal{G}} : \mathcal{F} \times \mathcal{G} \longrightarrow \mathcal{G},$$

and similarly a projection into $\mathcal{F}$. Let $\mathrm{pr}_{\mathcal{G},g} : G \longrightarrow F \cup G$ be the inclusion, and $\mathrm{pr}_{\mathcal{G},s}[b] := \mathrm{id}$ for all $b \in V_G$. Observe that $\mathrm{pr}^*_{\mathcal{G}} : k^{n+m} = k^n \times k^m \longrightarrow k^m$ is the projection. Finally let $\widetilde{\mathrm{pr}}_{\mathcal{G}(l)}(j) := (\text{length of } \alpha) + j$. To verify that in this way we indeed obtain a morphism of SDS, observe that we have commutative diagrams

$$\begin{tikzcd} k^n \times k^m \arrow[r, "f_a \times \mathrm{id}"] \arrow[d, "\mathrm{pr}^*_{\mathcal{G}}"'] & k^n \times k^m \arrow[d, "\mathrm{pr}^*_{\mathcal{G}}"] \\ k^m \arrow[r, "\mathrm{id}"] & k^m \end{tikzcd} \qquad \begin{tikzcd} k^n \times k^m \arrow[r, "\mathrm{id} \times g_b"] \arrow[d, "\mathrm{pr}^*_{\mathcal{G}}"'] & k^n \times k^m \arrow[d, "\mathrm{pr}^*_{\mathcal{G}}"] \\ k^m \arrow[r, "g_b"] & k^m \end{tikzcd} .$$

This shows that the second condition of a morphism is satisfied and that $\mathrm{pr}_{\mathcal{G}}$ is indeed a morphism of SDS.

It remains to verify the universal property of the product. Suppose we are given an SDS $\mathcal{K} = (K, (k[d]), (k_d), \delta)$ and morphisms $\varphi : \mathcal{K} \longrightarrow \mathcal{F}$ and $\psi : \mathcal{K} \longrightarrow \mathcal{G}$. We need to show that there is a unique morphism

$$\omega : \mathcal{K} \longrightarrow \mathcal{F} \times \mathcal{G},$$

such that $\mathrm{pr}_{\mathcal{F}} \circ \omega = \varphi$ and $\mathrm{pr}_{\mathcal{G}} \circ \omega = \psi$. Define

$$\omega_g : F \cup G \longrightarrow K$$

to be equal to φ_g, resp. ψ_g, on the component F, resp. G. Define $\omega_s[c] := \varphi_s[a]$ for $c = a \in V_F$ and $\omega_s[c] := \psi_s[b]$ for $c = b \in V_G$. Since a connected component of $\mathcal{F} \times \mathcal{G}$ is a connected component of either $\mathcal{F}$ or $\mathcal{G}$ the order preserving map $\widetilde{\omega}_{(l)}$ is determined by either $\widetilde{\varphi}_{(l)}$ or $\widetilde{\psi}_{(l)}$.

Then clearly $(\mathrm{pr}_{\mathcal{F}}\circ\omega)_g = \varphi_g$ and $(\mathrm{pr}_{\mathcal{G}}\circ\omega)_g = \psi_g$. Furthermore $(\mathrm{pr}_{\mathcal{F}}\circ\omega)_s[a] = \mathrm{pr}_{\mathcal{F},s}[a]\omega_s[\mathrm{pr}_{\mathcal{F},g}(a)] = \omega_s[a] = \varphi_s[a]$ for all $a \in V_F$ and similarly $(\mathrm{pr}_{\mathcal{G}} \circ \omega)_s[b] = \psi_s[b]$.

It is clear now that ω is a morphism uniquely determined by φ and ψ. □

Lemma 4.2. *Let $\mathcal{F} = (F, (k[a]), (f_a), \alpha)$ be an SDS. If α_i and α_{i+1} are in different connected components of F, then*

$$(F, (k[a]), (f_a), (\alpha_1, \dots, \alpha_{i+1}, \alpha_i, \dots, \alpha_r))$$

is isomorphic to $\mathcal{F}$.

Proof. Define $\varphi : \mathcal{F} \longrightarrow (F, (k[a]), (f_a), (\alpha_1, \dots, \alpha_{i+1}, \alpha_i, \dots, \alpha_r))$. Use $\varphi_g := (\mathrm{id} : F \longrightarrow F)$ and $\varphi_s[a] = \mathrm{id}$. Let $\alpha_i \in F_{(1)}$ and $\alpha_{i+1} \in F_{(2)}$. Then let $\widetilde{\varphi}_{(2)}(i) := i + 1$ and $\widetilde{\varphi}_{(1)}(i + 1) := i$ and $\widetilde{\varphi}_{(l)} := \mathrm{id}$ otherwise. This is obviously a canonical isomorphism. □

Thus the update schedule α of any SDS $\mathcal{F}$ may be rearranged according to the connected components of F and this gives a canonically isomorphic *SDS*.

Theorem 4.3. (1) *An SDS is indecomposable (w.r.t. products) if and only if the underlying graph is connected.*
(2) *Any SDS is uniquely isomorphic to the product of its connected components, and the connected components are uniquely determined.*

Proof. If $\mathcal{F}$ is a proper product then the underlying graph F is not connected. If F has more than one connected component, then by the preceding lemma it is canonically isomorphic to the product of its connected components as constructed above. □

To study morphisms of SDS it suffices to know the morphisms of the form $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ where $\mathcal{G}$ is indecomposable or connected. This holds since

$$\mathrm{Mor}(\mathcal{F}, \mathcal{G}_{(1)} \times \ldots \times \mathcal{G}_{(r)}) \cong \prod \mathrm{Mor}(\mathcal{F}, \mathcal{G}_{(i)}).$$

5. Decomposition of Morphisms

Theorem 5.1. *Let $\mathcal{F} = \mathcal{F}_1 \times \ldots \times \mathcal{F}_n$ with indecomposable components $\mathcal{F}_i$ and let $\mathcal{G}$ be indecomposable. Let $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ be a morphism. Then there is a uniquely determined component $\mathcal{F}_i$ and a uniquely determined morphism $\varphi_i : \mathcal{F}_i \longrightarrow \mathcal{G}$ such that*

$$(\varphi : \mathcal{F} \longrightarrow \mathcal{G}) = (\varphi_i \mathrm{pr}_i : \mathcal{F} \longrightarrow \mathcal{F}_i \longrightarrow \mathcal{G}).$$

Therefore

$$\mathrm{Mor}(\mathcal{F}_1 \times \ldots \times \mathcal{F}_n, \mathcal{G}) \cong \bigcup_{i=1}^{n} \mathrm{Mor}(\mathcal{F}_i, \mathcal{G}).$$

Proof. These assertions follow from the construction of the product and the fact that the image of a connected graph is connected. □

An indecomposable SDS must be considered in some sense as an autonomous system. So an SDS may be considered as a parallel system of several connected components. The above theorem also implies that every morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$, where $\mathcal{F}$ and $\mathcal{G}$ are arbitary SDS, can be described by a family of morphisms $\varphi_j : \mathcal{F}_{i_j} \longrightarrow \mathcal{G}_j$, where the $\mathcal{F}_{i_j}$ are suitable indecomposable components of $\mathcal{F}$, and the $\mathcal{G}_j$ run through all r indecomposable components of $\mathcal{G}$. So we could write

$$\varphi = (\varphi_1, \dots, \varphi_r).$$

Therefore it is sufficient only to study morphisms between indecomposable SDS.

6. Equalizers

Example 6.1. We want to give an example of a nontrivial equalizer in the category **SDS**. This will also give us some examples and show the great variety of morphisms between fairly small SDS.

Let $\mathcal{G} := \big((b), (k), (g_b = \mathrm{id}_b), \beta = (b, b)\big)$ be an SDS on the one vertex graph. The local and global update function is the identity $\mathrm{id} : k \longrightarrow k$. Finally the update schedule is a two letter word (b, b).

Let $\mathcal{H} := \big((b), (k), (h_b = \tau), \gamma = (b, b)\big)$ be an SDS with local function $\tau : k \longrightarrow k$, the transposition $\tau(0) = 1, \tau(1) = 0$.

We construct two morphisms $\varphi, \varphi' : \mathcal{G} \longrightarrow \mathcal{H}$ by

$$\begin{array}{lll} \varphi_g(b) = b, & \widetilde{\varphi}(1) = 1, & \widetilde{\varphi}(2) = 1; \\ \varphi'_g(b) = b, & \widetilde{\varphi}'(1) = 2, & \widetilde{\varphi}'(2) = 2. \end{array}$$

Observe that the numbers 1 and 2 are the indices or positions of the letters in the word (b, b). Furthermore we use $\varphi_s[b] = {\varphi'}_s[b] := \mathrm{id} : k \longrightarrow k$.

In order to check that these are morphisms we only have to show the second property of morphisms, namely that the two diagrams

$$\begin{tikzcd} k \arrow[r, "g_b=\mathrm{id}"] \arrow[d, "\varphi^*=\mathrm{id}"'] & k \arrow[d, "\varphi^*=\mathrm{id}"] \\ k \arrow[r, "h_b^2=\mathrm{id}"] & k \end{tikzcd} \qquad \begin{tikzcd} k \arrow[r, "g_b=\mathrm{id}"] \arrow[d, "\varphi^*=\mathrm{id}"'] & k \arrow[d, "\varphi^*=\mathrm{id}"] \\ k \arrow[r, "\mathrm{id}"] & k \end{tikzcd}$$

commute. The first diagram arises from the counterimage of the first letter of $\beta = (b, b)$ consisting of two instances of the letter b, and the second diagram arises from the counterimage of the second letter of $\beta = (b, b)$ that is empty. Similarly we show for φ' that the diagrams

$$\begin{tikzcd} k \arrow[r, "g_b=\mathrm{id}"] \arrow[d, "\varphi'^*=\mathrm{id}"'] & k \arrow[d, "\varphi'^*=\mathrm{id}"] \\ k \arrow[r, "\mathrm{id}"] & k \end{tikzcd} \qquad \begin{tikzcd} k \arrow[r, "g_b=\mathrm{id}"] \arrow[d, "\varphi'^*=\mathrm{id}"'] & k \arrow[d, "\varphi'^*=\mathrm{id}"] \\ k \arrow[r, "h_b^2=\mathrm{id}"] & k \end{tikzcd}$$

commute.

Let $\mathcal{K} = \big((b), (k), (l_b = \mathrm{id}), (b)\big)$ be an SDS and define a morphism $\iota : \mathcal{K} \longrightarrow \mathcal{G}$ by $\iota(b) = b$ (reverse direction!) and $\widetilde{\iota}(1) = 1 = \widetilde{\iota}(2)$. Since the diagram

$$\begin{tikzcd} k \arrow[r, "l_b=\mathrm{id}"] \arrow[d, "\iota^*=\mathrm{id}"'] & k \arrow[d, "\iota^*=\mathrm{id}"] \\ k \arrow[r, "g_b^2=\mathrm{id}"] & k \end{tikzcd}$$

commutes, ι is a morphism and we have morphisms

$$\mathcal{K} \xrightarrow{\iota} \mathcal{G} \underset{\varphi'}{\overset{\varphi}{\rightrightarrows}} \mathcal{H}\ .$$

We have $\varphi\iota = \varphi'\iota$ since $\mathcal{K}$ has only one element components. We claim that $(\mathcal{K}, \iota)$ is an equalizer of the pair (φ, φ').

Let $\mathcal{F} = \big(F, (k[a]), (f_i), \alpha\big)$ be an SDS and $\psi : \mathcal{F} \longrightarrow \mathcal{G}$ be a morphism such that $\varphi\psi = \varphi'\psi$. We have to show that there is a unique morphism $\nu : \mathcal{F} \longrightarrow \mathcal{K}$ such that $\iota\nu = \psi$. Let $\psi_g(b) := a \in V_F$, and let $\widetilde{\psi}(1) = i$ and $\widetilde{\psi}(2) = j$, i.e. $\alpha_i = a = \alpha_j$. From $\varphi\psi = \varphi'\psi$ we get $\widetilde{\psi}\widetilde{\varphi} = \widetilde{\psi}\widetilde{\varphi}'$ and hence $i = \widetilde{\psi}\widetilde{\varphi}(1) = \widetilde{\psi}\widetilde{\varphi}'(1) = j$. Furthermore we have $\psi^* = \psi_s[b]\mathrm{pr}_a : k^n \longrightarrow k[a] \longrightarrow k[b]$, and we have commutative diagrams

$$\begin{array}{ccc} k^n & \xrightarrow{f_a} & k^n \\ \downarrow{\scriptstyle \psi^*} & & \downarrow{\scriptstyle \psi^*} \\ k & \xrightarrow{g_b = \mathrm{id}} & k \end{array} \qquad \text{and} \qquad \begin{array}{ccc} k^n & \xrightarrow{f_{a'}} & k^n \\ \downarrow{\scriptstyle \psi^*} & & \downarrow{\scriptstyle \psi^*} \\ k & \xrightarrow{\mathrm{id}} & k \end{array}$$

for all $a' \neq a$ in F.

Now define $\nu : \mathcal{F} \longrightarrow \mathcal{K}$ by $\nu_g(b) := a$, $\nu_s[b] = \psi_s[b]$, and $\widetilde{\nu}(1) := i = j$. This is obviously the only choice if we want to get $\iota\nu = \psi$. Then we have $\nu^* = \nu_s[b]\mathrm{pr}_a : k^n \longrightarrow k$ and the diagrams

$$\begin{array}{ccc} k^n & \xrightarrow{f_a} & k^n \\ \downarrow{\scriptstyle \nu^*} & & \downarrow{\scriptstyle \nu^*} \\ k & \xrightarrow{l_a = \mathrm{id}} & k \end{array} \qquad \begin{array}{ccc} k^n & \xrightarrow{f_{a'}} & k^n \\ \downarrow{\scriptstyle \nu^*} & & \downarrow{\scriptstyle \nu^*} \\ k & \xrightarrow{\mathrm{id}} & k \end{array}$$

commute, hence ι is the unique morphism such that $\iota\nu = \psi$. So $(\mathcal{K}, \iota)$ is an equalizer.

For the next two remarks we will assume $Z = \{k, \mathrm{id}_k\}$.

Remark 6.2. We want to show that, in general, there are no equalizers in the category of SDS. Let $\mathcal{G} = (G, (k[b]), (g_b), \beta)$ and $\mathcal{H} = (H, (k[c]), (h_c), \gamma)$ be the following SDS

- $V_G := \{a, b, c, d\}$, $E_G := \{\{a, b\}, \{c, d\}\}$,
- $k[b] := k = \{0, 1\}$ for all $b \in V_G$,
- $g_b :=$ id for all $b \in V_G$,
- $\beta := (a, b, c, d) = (b_1, b_2, b_3, b_4)$.

- $V_H := \{a\}$, $E_F := \emptyset$,
- $k[c] := k = \{0, 1\}$ for all $c \in V_H$,
- $h_c :=$ id for all $c \in V_H$,
- $\gamma := (a) = (c_1)$.

Let $\varphi, \psi : \mathcal{G} \longrightarrow \mathcal{H}$ be two morphisms given by

- $\varphi_g(a) := a$, $\psi_g(a) := c$,
- $\varphi_s[a] := \mathrm{id} =: \psi_s[a]$,
- $\widetilde{\varphi}(1) := 1$, $\widetilde{\psi}(1) := 3$.

We want to find an equalizer $\rho : \mathcal{E} \longrightarrow \mathcal{G}$ of these two morphisms (i.e. $\varphi\rho = \psi\rho$ and $(\mathcal{E}, \rho)$ universal w.r.t. this property).

As a test object we use the SDS $\mathcal{F} = (F, (k[a]), (f_a), \alpha)$ defined as follows

- $V_F := \{a, b, d\}$, $E_F := \{\{a, b\}, \{a, d\}\}$,
- $k[a] := k = \{0, 1\}$ for all $a \in V_F$,
- $f_a :=$ id for all $a \in V_F$,
- $\alpha := (a, b, d) = (a_1, a_2, a_3)$.

The following two morphisms $\sigma, \tau : \mathcal{F} \longrightarrow \mathcal{G}$ should serve as test morphisms. They are given by

- $\sigma_g(a) = \sigma_g(c) := a$, $\sigma_g(b) := b$, $\sigma_g(d) = d$,
  $\tau_g(a) = \tau_g(c) := a$, $\tau_g(b) := d$, $\tau_g(d) = b$,
- $\sigma_s[a] :=$ id $=: \tau_s[a]$ for all $a \in V_G$,
- $\widetilde{\sigma}(1) := 1$, $\widetilde{\sigma}(2) := 2$, $\widetilde{\sigma}(3) := 1$, $\widetilde{\sigma}(4) := 3$,
  $\widetilde{\tau}(1) := 1$, $\widetilde{\tau}(2) := 3$, $\widetilde{\tau}(3) := 1$, $\widetilde{\tau}(4) := 2$.

Then it is easy to see that $\sigma\varphi = \sigma\psi$ and $\tau\varphi = \tau\psi$.

Assume that we have an equalizer $\rho : \mathcal{E} \longrightarrow \mathcal{G}$ of φ and ψ. Then ρ_g must have the following images $\rho_g(a) = \rho_g(c) =: \bar{a}$ (since $\rho_g\varphi_g = \rho_g\psi_g$), $\rho_g(b) =: \bar{b}$, and $\rho_g(d) =: \bar{d}$ in V_E. Furthermore let $\widetilde{\rho}(1) =: i_1$, $\widetilde{\rho}(2) =: i_2$, $\widetilde{\rho}(3) =: i_3$, and $\widetilde{\rho}(4) =: i_4$. Then we have $i_1 = \widetilde{\rho}(1) = \widetilde{\rho}\widetilde{\varphi}(1) = \widetilde{\rho}\widetilde{\psi}(1) = \widetilde{\rho}(3) = i_3$. Since $\widetilde{\rho}$ is order preserving on the connected components, we get $i_1 \leq i_2$ and $i_1 \leq i_4$. Since all of i_1, i_2, i_4 map into a connected component of the graph E (they map into $\bar{a}, \bar{b}, \bar{d}$ resp.) we must have i_2 and i_4 comparable in the order of $|\delta|$, the update schedule of $\mathcal{E}$.

The morphisms σ and τ have unique factorizations $\sigma = \rho\sigma_0$ and $\tau = \rho\tau_0$ through the equalizer. Now $\widetilde{\sigma}_0(i_2) = \widetilde{\sigma}(2) = 2$ and $\widetilde{\sigma}_0(i_4) = \widetilde{\sigma}(4) = 3$. Since $\widetilde{\sigma}_0$ is order preserving we get $i_2 < i_4$.

With the same argument for τ we get $i_2 > i_4$ a contradiction. Hence there cannot exist an equalizer for φ, ψ.

Remark 6.3. In this context and with $Z = \{k, \mathrm{id}_k\}$ it might appear as if $\mathcal{O} = \big((a), (k), (\mathrm{id}), (a)\big)$ is an initial object in **SDS**. This is not the case since this SDS does not admit a morphism into any SDS with update functions not the identity on the diagonal in k^n, the diagram

$$\begin{array}{ccc} k & \xrightarrow{\mathrm{id}} & k \\ {\scriptstyle\Delta}\downarrow & & \downarrow{\scriptstyle\Delta} \\ k^n & \xrightarrow{f_\alpha} & k^n \end{array}$$

does not commute. To complete our study of products we have, however, the empty product.

Lemma 6.4. *The SDS* $\big(\emptyset, \emptyset = (), \emptyset = (), \emptyset = ()\big)$ *is a final object in* **SDS**.

Proof. There is a unique morphism of graphs $\emptyset \longrightarrow F$ and the diagram

$$\begin{array}{ccc} k^n & \xrightarrow{f_i} & k^n \\ \downarrow & & \downarrow \\ \{*\} & \xrightarrow{\mathrm{id}} & \{*\} \end{array}$$

commutes. □

7. Simulations and their Effects on State Spaces

As we mentioned in the introduction, we consider a morphism $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ as a simulation if φ is a monomorphism – then $\mathcal{G}$ is simulated by $\mathcal{F}$ – or, if φ is an epimorphism – then $\mathcal{F}$ is simulated by $\mathcal{G}$. We will only consider those monomorphisms φ where φ_g is surjective on the set of vertices, and the $\varphi_s[b]$ are injective. We will call them *injective monomorphisms.*

Lemma 7.1. *If $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ is an injective monomorphism, then $\varphi^* : k^n \longrightarrow k^m$ is an injective map.*

Proof. Let $\varphi^*(x_1, \ldots, x_n) = \varphi^*(y_1, \ldots, y_n)$. Then

$$\begin{aligned}&\big(\varphi_s[b_1](x[\varphi_g(b_1)]), \ldots, \varphi_s[b_m](x[\varphi_g(b_m)])\big)\\ &= \big(\varphi_s[b_1](y[\varphi_g(b_1)]), \ldots, \varphi_s[b_m](y[\varphi_g(b_m)])\big).\end{aligned}$$

Hence, for all $i = 1, \ldots, m$, we get $\varphi_s[b_i](x[\varphi_g(b_i)]) = \varphi_s[b_i](y[\varphi_g(b_i)])$. Since the $\varphi_s[b_i]$ are injective we get that $x[\varphi_g(b_i)] = y[\varphi_g(b_i)]$. Now, φ_g is surjective, so we get $(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$. □

Considering epimorphisms, we will only consider those $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$, where φ_g is injective on the set of vertices, and the $\varphi_s[b]$ are surjective. We will call them *surjective epimorphisms.*

Lemma 7.2. *If $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ is a surjective epimorphism, then $\varphi^* : k^n \longrightarrow k^m$ is a surjective map.*

Proof. Let $(y[b_1], \ldots, y[b_m]) \in k^m$. Then there is an m-tuple

$$\big(z[\varphi_g(b_1)], \ldots, z[\varphi_g(b_m)]\big) \in k[\varphi_g(b_1)] \times \ldots \times k[\varphi_g(b_m)],$$

such that $\big(\varphi_s[b_1](z[\varphi_g(b_1)]), \ldots, \varphi_s[b_m](z[\varphi_g(b_m)])\big) = (y[b_1], \ldots, y[b_m])$, since the $\varphi_s[b_i]$ are surjective. Define $x[a_i] := z[\varphi_g(b_j)] \in k[a_i]$, if $\varphi_g(b_j) = a_i$, and $x[a_i]$ arbitrary if $a_i \notin \mathrm{Im}(\varphi_g)$. The $x[a_i] = z[\varphi_g(b_j)]$ are well defined since φ_g is injective, so that there is a $(x[a_1], \ldots, x[a_n])$, such that

$$\big(x[\varphi_g(b_1)], \ldots, x[\varphi_g(b_m)]\big) = \big(z[\varphi_g(b_1)], \ldots, z[\varphi_g(b_m)]\big),$$

and thus φ^* is surjective. □

Proposition 7.3. *Let $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ be an injective monomorphism, and let $\mathcal{S}(\varphi) : \mathcal{S}(\mathcal{F}) \longrightarrow \mathcal{S}(\mathcal{G})$ be the graph morphism of the associated state spaces. Then*

(1) *$\mathcal{S}(\varphi)$ maps each limit cycle of $\mathcal{S}(\mathcal{F})$ bijectively onto a limit cycle of $\mathcal{S}(\mathcal{G})$;*
(2) *$\mathcal{S}(\varphi)$ is injective on the set of limit cycles;*
(3) *$\mathcal{S}(\varphi)$ maps transients injectively into transients, preserving endpoints.*

Proof. The three statements are clear, since $\mathcal{S}(\varphi)$ is injective on the set of vertices. □

Proposition 7.4. *Let $\varphi : \mathcal{F} \longrightarrow \mathcal{G}$ be a surjective epimorphism, and let $\mathcal{S}(\varphi) : \mathcal{S}(\mathcal{F}) \longrightarrow \mathcal{S}(\mathcal{G})$ be the graph morphism of the associated state spaces. Then*

(1) *$\mathcal{S}(\varphi)$ maps each limit cycle of length n of $\mathcal{S}(\mathcal{F})$ onto a limit cycle of length t of $\mathcal{S}(\mathcal{G})$, where t divides n;*
(2) *$\mathcal{S}(\varphi)$ maps the set of limit cycles of $\mathcal{S}(\mathcal{F})$ onto the set of limit cycles of $\mathcal{S}(\mathcal{G})$;*

(3) *if two nodes of a transient in $\mathcal{S}(\mathcal{F})$ of distance $n+1$ are mapped into the same node of $\mathcal{S}(\mathcal{G})$, then the path between the two nodes in $\mathcal{S}(\mathcal{F})$ is mapped onto a limit cycle of length t in $\mathcal{S}(\mathcal{G})$, where t divides n.*

Proof. Again, the three statements follow from the fact that $\mathcal{S}(\varphi) : \mathcal{S}(\mathcal{F}) \to \mathcal{S}(\mathcal{G})$ is surjective on the set of vertices. □

The two propositions show what kind of information about the dynamical behavior of some SDS is preserved if it is simulated by another SDS.

References

[1] C. L. Barrett, R. Thord, and C. M. Reidys, Knowledge and Networks in a Dynamical Economy, in *Simulations in Decision Making for Socio-technical Systems*, M. Beckman and B. Johansson and F. Snickars and R. Thord (eds.), Springer Verlag, New York, 1998.

[2] C. L. Barrett, and C. M. Reidys, Elements of a Theory of Simulation, I: Sequential CA Over Random Graphs, *Appl. Math. and Comput.* **98**, pp. 241–259, 1999.

[3] C. L. Barrett, H. S. Mortveit, and C. M. Reidys, Elements of a Theory of Simulation II: Sequential Dynamical Systems, *Appl. Math. and Comp.* **107** (2–3), pp. 121–136, 1999.

[4] C. L. Barrett, H. S. Mortveit, and C. M. Reidys, Elements of a Theory of Simulation III: Equivalence of SDS, *Appl. Math. and Comp.* **122**, pp. 325–340, 2001.

[5] L. Garcia, A. Jarrah, R. Laubenbacher, A Classification of Finite Dynamical Systems, preprint.

[6] R. Laubenbacher and B. Pareigis, Equivalence Relations on Finite Dynamical Systems, *Adv. in Appl. Math.* **26** (2001) 237–251.

Virginia Bioinformatics Institute, Blacksburg, VA, 24061

E-mail address: `reinhard@vbi.vt.edu`

Mathematisches Institut, Universität München, Theresienstr. 39, D-80333 München, Germany

E-mail address: `pareigis@rz.mathematik.uni-muenchen.de`