A TOUR OF REACTION SYSTEMS

ROBERT BRIJDER[1], ANDRZEJ EHRENFEUCHT[2], MICHAEL MAIN[2],
and GRZEGORZ ROZENBERG[1,2]

[1]*Leiden Institute of Advanced Computer Science, Leiden University*
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
rbrijder@liacs.nl and *rozenber@liacs.nl*

[2]*Department of Computer Science, University of Colorado at Boulder*
430 UCB, Boulder, CO 80309, USA
andrzej@colorado.edu and *main@colorado.edu*

Reaction systems are a formal framework for investigating processes carried out by biochemical reactions. This paper is an introduction to reaction systems. It provides basic notions together with the underlying intuition and motivation as well as two examples (a binary counter and transition systems) of "programming" with reaction systems. It also provides a tour of some research themes.

Keywords: natural computing, biochemical reactions, reaction systems, transition systems

1991 Mathematics Subject Classification: 68Q05, 68Q19, 68Q87, 68R05, 92C40

1. Introduction

Natural Computing is concerned with human-designed computing inspired by nature as well as with computation taking place in nature. The latter strand of research investigates in terms of information processing, processes taking place in nature, and the computational nature of biochemical reactions is a research theme belonging here.

Reaction systems are a formal framework for investigating processes carried out by biochemical reactions in living cells. The underlying idea here is that the interactions between biochemical reactions as well as the functioning of single reactions are based on the mechanisms of facilitation and inhibition.

This paper, based on the invited lecture at DLT 2010 [6], is an introduction to reaction systems directed at researchers in theoretical computer science. It first provides the basic notions related to reaction systems together with the underlying intuition and motivation (based on the biochemistry of living cells). To provide a practice of basic notions, we then discuss two examples of "programming" with

reaction systems: an implementation of a binary counter and an implementation of a finite, deterministic transition system.

The paper then continues with a short tour of three recent research themes. We hope that the three sections corresponding to these themes, together with the discussion section, will provide the reader with a "taste" of theoretical research in reaction systems.

2. Preliminaries

In this paper we use standard mathematical terminology and notation. To avoid misunderstandings, we recall some notions and notation.

We use $\mathbb{Z}^+$ to denote the set of positive integers. Unless otherwise clear, we consider finite sets only. The cardinality of a set X is denoted by $|X|$, and the powerset of X is denoted by 2^X. The symmetric difference of sets Z_1 and Z_2, defined by $(Z_1 \setminus Z_2) \cup (Z_2 \setminus Z_1)$, is denoted by $Z_1 \oplus Z_2$. The composition of functions f and g is denoted by gf (first f, then g). Functions, $f : X \to Y$ and $g : U \to V$, are *isomorphic* provided that there exist bijections $p : X \to U$ and $h : Y \to V$ such that $hf = gp$.

A *directed graph* (*digraph*) is given by an ordered pair (V, E), where V is the set of vertices and E is the set of directed edges.

Let $\tau = W_0, W_1, \cdots, W_n$ be a sequence of sets. For a set Q, the Q-*projection of* τ is the sequence of sets $\mathrm{proj}_Q(\tau) = W_0 \cap Q, W_1 \cap Q, \cdots, W_n \cap Q$.

3. Reactions and Reaction Systems

In this section we recall the basic notions concerning reaction systems (see, e.g., [3, 4, 5]). A biochemical reaction can take place if all of its reactants are present in a given state and none of its inhibitors is present; when a reaction takes place, it creates its products. This intuition leads to the following definition.

Definition 1. *A* reaction *is a triplet $a = (R, I, P)$, where R, I, P are finite nonempty sets with $R \cap I = \varnothing$. If S is a set such that $R, I, P \subseteq S$, then a is a* reaction in S.

Note that both the reactant set and the inhibitor set are required to be non-empty so that all products are produced from at least one reactant and that every reaction can be inhibited in some way. The notion of a reaction without an inhibitor can be simulated by introducing an artificial inhibitor that is never produced by any reaction.

The sets R, I, P are also written R_a, I_a, P_a, and called the *reactant set of a*, the *inhibitor set of a*, and the *product set of a*, respectively. Also, $M_a = R_a \cup I_a$ is the set of *resources of a*, $m_a = |M_a|$, and $\mathrm{rac}(S)$ denotes the set of all reactions in S. Note that since R_a and I_a are both nonempty, we always have $|M_a| \geq 2$.

Example 2. *Let* $S = \{1, 2, 3, 4\}$ *and* $a = (\{3\}, \{1, 2\}, \{1, 2, 4\})$. *Then* $a \in \mathrm{rac}(S)$ *and* $M_a = \{1, 2, 3\}$.

Definition 3. *Let* T *be a finite set.*

(1) *Let* a *be a reaction. Then* a *is* enabled by T*, denoted by* $en_a(T)$*, if* $R_a \subseteq T$ *and* $I_a \cap T = \varnothing$*. The* result of a on T*, denoted by* $\mathrm{res}_a(T)$*, is defined by:* $\mathrm{res}_a(T) = P_a$ *if* $en_a(T)$*, and* $\mathrm{res}_a(T) = \varnothing$ *otherwise.*

(2) *Let* A *be a finite set of reactions. The* result of A on T*, denoted by* $\mathrm{res}_A(T)$*, is defined by:* $\mathrm{res}_A(T) = \bigcup_{a \in A} \mathrm{res}_a(T)$.

The intuition behind T is that of a state of a biochemical system, i.e., a set of biochemical entities present in the current biochemical environment. Thus a is enabled by T if T separates R_a from I_a, i.e., $R_a \subseteq T$ and $I_a \cap T = \varnothing$. The result of a set of reactions A on T is cumulative, i.e., it is the union of results of all individual reactions from A. In fact, $\mathrm{res}_A(T) = \bigcup \{\mathrm{res}_a(T) \mid a \in A \text{ and } en_a(T)\}$.

Note that if $a, b \in A$ with both a and b enabled by T, then even if $R_a \cap R_b \neq \varnothing$, still both $P_a \subseteq \mathrm{res}_A(T)$ and $P_b \subseteq \mathrm{res}_A(T)$. Hence there is no conflict of resources here—both a and b use $R_a \cap R_b$ to produce their products. Thus the nonempty intersection of sets of reactants of enabled reactions does not constitute a *conflict*, which is a major difference with standard models of concurrent systems, such as Petri nets [10]. This setup reflects our *threshold assumption*: Either a resource is present and then it is present in a "sufficient amount," or it is not present. Consequently, there is *no counting* in reaction systems: One deals here with sets rather than with multisets.

Example 4. *Consider again* S *and* a *from Example 2. Let* $T_1 = \{3, 4\}$ *and* $T_2 = \{2, 3, 4\}$*. Since* $en_a(T_1)$*, we have* $\mathrm{res}_{\{a\}}(T_1) = P_a = \{1, 2, 4\}$*. On the other hand,* a *is not enabled on* T_2*, and so* $\mathrm{res}_{\{a\}}(T_2) = \varnothing$*. Let now* $b = (\{3, 4\}, \{2\}, \{3\})$ *be another reaction. Then* $\mathrm{res}_{\{a,b\}}(T_1) = P_a \cup P_b = \{1, 2, 3, 4\}$*. Note that although* $R_a \cap R_b \neq \varnothing$*, both reactions are enabled in* T_1 *and produce their products (there is no conflict).*

We are ready now to recall the notion of a reaction system, which essentially is just a finite set of reactions.

Definition 5. *A* reaction system*, abbreviated rs, is an ordered pair* $\mathcal{A} = (S, A)$ *such that* S *is a finite set, and* $A \subseteq \mathrm{rac}(S)$.

The set S is called the *background set* of $\mathcal{A}$, its elements are called *entities*—they represent molecular entities (e.g., atoms, ions, molecules) that may be present in the states of a biochemical system modeled by $\mathcal{A}$. The set A is called the *set of reactions of* $\mathcal{A}$—note that since S is finite, so is A.

The dynamic behavior of a rs is formalized through the notion of an interactive process.

Example 7. March 13, 2011 10:31
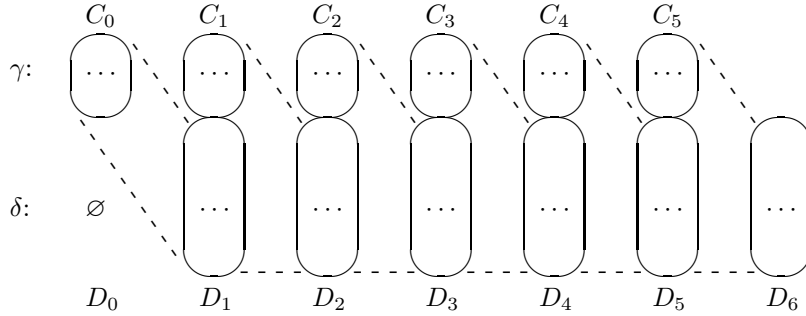BRIJDER/EHRENFEUCHT/MAIN/ROZENBERG rst

Fig. 1. Interactive process of a reaction system

Definition 6. *Let* $\mathcal{A} = (S, A)$ *be a rs and let* $n \geq 0$ *be an integer. An* (n-step) *interactive process in* $\mathcal{A}$ *is a pair* $\pi = (\gamma, \delta)$ *of finite sequences such that* $\gamma = C_0, \cdots, C_n$ *and* $\delta = D_0, \cdots, D_n$, *where* $C_0, \cdots, C_n, D_0, \cdots, D_n \subseteq S$, $D_0 = \varnothing$, *and* $D_i = \mathrm{res}_{\mathcal{A}}(D_{i-1} \cup C_{i-1})$ *for all* $i \in \{1, \cdots, n\}$.

The notion of an interactive process is visualized in Figure 1. Note that although in the figure each pair (C_i, D_i) is shown as a pair of disjoint ovals, by definition of an interactive process, C_i and D_i do not have to be disjoint. In the figure we show C_i and D_i as disjoint ovals since otherwise (with their intersection shown) the illustration would become too involved, especially when we move to more complex processes in Section 6.

The sequence γ is the *context sequence of* π, denoted by $\mathrm{con}(\pi)$, and the sequence δ is the *result sequence of* π, denoted by $\mathrm{res}(\pi)$. Then the sequence $\tau = W_0, W_1, \cdots, W_n$ defined by $W_i = C_i \cup D_i$ for all $i \in \{0, \cdots, n\}$ is the *state sequence of* π, denoted by $\mathrm{st}(\pi)$, with $W_0 = C_0$ called the *initial state of* π (and of τ) denoted $init(\pi)$ (and $init(\tau)$). If $C_i \subseteq D_i$ for all $i \in \{1, \cdots, n\}$, then we say that π (and τ) are *context-independent*. Note that for any n-step, context-independent interactive process, we can take $C_i = \varnothing$ for all $i \in \{1, \cdots, n\}$ without changing the state sequence.

The context sequence formalizes the intuition that, in general, a rs is not a closed system, and so its behavior is influenced by its "environment." Note that a context-independent state sequence depends only on the initial state W_0 and its length $(n+1)$. Also, in a context-independent state sequence $\tau = W_0, \cdots, W_i, W_{i+1}, \cdots, W_n$, during the transition from W_i to W_{i+1} all entities from $W_i - \mathrm{res}_{\mathcal{A}}(W_i)$ vanish. This reflects our assumption of *no permanency*: An entity from a current state vanishes unless it is produced/sustained by $\mathcal{A}$. Clearly, if τ is not context-independent, then an entity from a current state can be also sustained (thrown in) by the context (C_{i+1}). This feature is also a major difference with standard models of concurrent systems such as Petri nets ([10]).

Consider the reaction system $\mathcal{U} = (\{1,2,3,4\}, U)$, *where U is the set of these six reactions:*

$$(\{1\}, \{3\}, \{2\}), \quad (\{2\}, \{1\}, \{1\}), \quad (\{2\}, \{3\}, \{3\}),$$
$$(\{3\}, \{1,2\}, \{1,2,4\}), \quad (\{4\}, \{3\}, \{1,2\}), \quad (\{1,3\}, \{2,4\}, \{2,3\}).$$

Then the sequence $\tau = \{1,2,4\}, \{1,2,3\}, \varnothing$ is a context-independent state sequence of $\mathcal{U}$. Notice that, e.g., $W_1 = D_1 \cup C_1 = \text{res}_{\mathcal{U}}(\{1,2,4\}) \cup \varnothing = \{1,2,3\}$.

The set of all state sequences of a reaction system $\mathcal{A}$ (i.e., all state sequences of all interactive processes in $\mathcal{A}$) is denoted by $STS(\mathcal{A})$, and the set of all context-independent state sequences of $\mathcal{A}$ is denoted by $CISTS(\mathcal{A})$.

4. Examples

4.1. *A binary counter*

A reaction system can act as a cyclic n-bit counter in which external signals trigger increment or decrement operations. To build the counter, let $n > 0$ be an integer and define the background set as

$$\{p_0, p_1, \cdots, p_{n-1}\} \cup \{dec, inc\}.$$

The entities p_0 to p_{n-1} are used to represent individual bits in an n-bit integer (in a way that we'll describe next); the *dec* entity is a request to decrement the represented number by one, and the *inc* entity is a request to increment by one.

The representation of an n-bit unsigned integer by a state $M \subseteq \{p_0, p_1, \cdots, p_{n-1}, dec, inc\}$ is defined by the individual entities in M. An entity $p_j \in M$ represents a one in the position corresponding to 2^j in an n-bit number; hence, $p_j \notin M$ represents a zero in this position. For example, with $n = 8$, the state $\{p_5, p_3, p_1\}$ represents 00101010 (*i.e.*, 42 in base 10).

Our goal is to have the *dec* entity cause the counter to decrement by one, and the *inc* entity causes an increment of one. Hence, the result of the reaction system on $\{p_5, p_3, p_1, dec\}$ will be $\{p_5, p_3, p_0\}$ (*i.e.*, 41 in base 10). Similarly, the result on $\{p_5, p_3, p_1, inc\}$ will be $\{p_5, p_3, p_1, p_0\}$ (*i.e.*, 43). To accomplish this behavior, the system has three sets of reactions described below.

(1) The first group of reactions causes bits to be retained in the next state when there are no decrement or increment requests.

Retention: For each j with $0 \le j < n$, define the reaction

$$a_j = (\{p_j\}, \{dec, inc\}, \{p_j\}).$$

(2) The second group of reactions implements the increment operation. As follows from binary arithmetic, when a number is incremented, the least significant zero bit is flipped to one. All the less significant bits are flipped to zeros, and higher bits are unchanged. The needed reactions are given below.

With an increment, flip the least significant zero to one:
For each j with $0 \leq j < n$, define the reaction

$$b_j = (\{inc, p_0, p_1, \cdots, p_{j-1}\}, \{dec, p_j\}, \{p_j\}).$$

Note that the case of $j = 0$ is the reaction

$$b_0 = (\{inc\}, \{dec, p_0\}, \{p_0\}).$$

The more significant bits remain:
For each j and k with $0 \leq j < k < n$, define the reaction

$$c_{j,k} = (\{inc, p_k\}, \{dec, p_j\}, \{p_k\}).$$

To analyze these rules, suppose that the least significant zero of an n-bit number is in the 2^j position, and an increment is requested. Bits that are less significant than the 2^j position will disappear because there are no enabled reactions to produce them. Reaction b_j will produce the 2^j bit. And for each k with $j < k < n$, the 2^k bits keep their original values. (In particular, the 2^k bit is produced by $c_{j,k}$ if and only if it was present in the original n-bit number.) Note that the counter is cyclic: If all bits are present in a state (*e.g.*, 11111), then an increment results in a wrap-around to zero (*e.g.*, 00000).

(3) The third group of reactions implements the decrement operation in a way that's similar to what we've already seen.

Flip zeros to ones when there is no one at a lower position:
For each j with $0 \leq j < n$, define the reaction

$$d_j = (\{dec\}, \{inc, p_0, p_1, \cdots, p_j\}, \{p_j\}).$$

The more significant bits remain:
For each j and k with $0 \leq j < k < n$, define the reaction

$$e_{j,k} = (\{dec, p_j, p_k\}, \{inc\}, \{p_k\}).$$

Note that the d and e rules work correctly when the 2^j bit is one and all less significant bits are zero (in which case the 2^j bit is not produced for the next state).

The decrement rules also have a notable cyclic behavior: If no bits are present (*e.g.*, 00000), then a decrement results in a wrap-around to all ones (*e.g.*, 11111). We can also ask about the system's behavior when both *dec* and *inc* are present. In this case, every reaction is inhibited, and the next state will be empty. Hence, this is a convenient way to request the counter to reset to zero.

We can now define the complete n-bit counter reaction system, $\mathcal{B}_n = (S_n, B_n)$, in which the reaction set, B_n, is the union of the reaction sets we have thus far presented:

$$S_n = \{p_0, p_1, \cdots, p_{n-1}\} \cup \{dec, inc\}, \text{ and}$$

$$B_n = \{a_j, b_j, d_j \mid 0 \leq j < n\} \cup \{c_{j,k}, e_{j,k} \mid 0 \leq j < k < n\}.$$

To illustrate the system in action, let $n = 4$ and consider the interactive process with this sequence of contexts:

$$C_0 = \{p_1, p_3\}, C_1 = \varnothing, C_2 = \{inc\}, C_3 = \{inc\}, C_4 = \{dec\}, C_5 = \{dec, inc\}.$$

As always, the first context, C_0, is the initial state of the process; in this case, it represents the binary number 1010. At steps 2 and 3, the context requests an increment; C_4 requests a decrement; and C_5 requests a reset. The complete interactive process from these contexts is shown stepwise in the following diagram with the corresponding binary number at the bottom of each column.
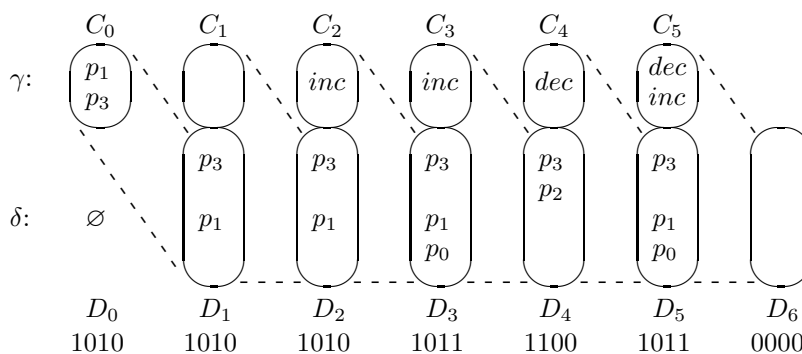
4.2. *Finite transition systems*

We have seen how the rules of binary increment and decrement translate easily into a reaction system. As a similar but more general example, we show how to translate from finite deterministic transition systems to reaction systems. We assume familiarity with the basic definitions and notation of deterministic finite transition systems. We use these specific definitions and notations:

Definition 8. *A deterministic, finite transition system is a 3-tuple, (Q, Σ, δ), where:*

    Q *is a finite, non-empty set of states,*
    Σ *is a finite, non-empty set of characters (the input alphabet),*
    $\delta : Q \times \Sigma \to Q$ *is the* state-transition function.

Definition 9. *Let $\mathbf{F} = (Q, \Sigma, \delta)$ be a finite transition system and let $n \geq 0$ be an integer. Consider a sequence of states $q_0, q_1, \cdots, q_n$ (with each $q_i \in Q$) and a string of characters $b_1...b_n$ (with each $b_i \in \Sigma$). If $\delta(q_{i-1}, b_i) = q_i$ for each i in the range*

8   *Brijder, Ehrenfeucht, Main, Rozenberg*

$1 \leq i \leq n$, *then we say that* $q_0, q_1, \cdots, q_n$ *is a* transition sequence *for the string* $b_1...b_n$, *and we write:*

$$q_0 \xrightarrow{b_1} q_1 \xrightarrow{b_2} \cdots \xrightarrow{b_n} q_n.$$

For our construction, we start with a deterministic finite transition system $\mathbf{F} = (Q, \Sigma, \delta)$ with Q and Σ disjoint and furthermore $|Q \cup \Sigma| > 2$. (We could remove these restrictions, but the construction is clearer with these requirements in place.)

Our goal is to build a reaction system $\mathcal{RS}_{\mathbf{F}} = (S_{\mathbf{F}}, A_{\mathbf{F}})$ in which the transition sequences of $\mathbf{F}$ are in one-to-one correspondence with a certain kind of interactive process of $\mathcal{RS}_{\mathbf{F}}$. In particular, $\mathbf{F}$ has the transition sequence

$$p_0 \xrightarrow{x_1} p_1 \xrightarrow{x_2} \cdots \xrightarrow{x_n} p_n$$

if and only if the constructed $\mathcal{RS}_{\mathbf{F}}$ has the interactive process shown here:
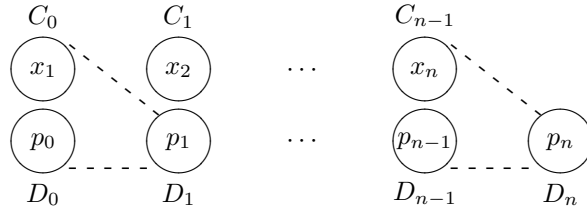
For clarity in this construction, we have departed from the assumption that D_0 is empty; if necessary elsewhere, D_0 could be made empty by placing p_0 into C_0.

The reaction system provides a natural representation of the transition system. In the interactive processes, the entities in the contexts are characters from the alphabet Σ, and the entities in the results are elements of Q. Hence, the background set of $\mathcal{RS}_{\mathbf{F}}$ is constructed as the union:

$$S_{\mathbf{F}} = Q \cup \Sigma.$$

The reactions of $\mathcal{RS}_{\mathbf{F}}$ come from the transitions of $\mathbf{F}$. In particular, suppose that $\delta(p, b) = q$ for some $p, q \in Q$ and $b \in \Sigma$, so that $\mathbf{F}$ has a transition shown here:
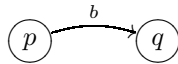
Then $\mathcal{RS}_{\mathbf{F}}$ has a reaction, $a_{p,b,q}$ defined by:

$$a_{p,b,q} = (\{p, b\}, S_{\mathbf{F}} - \{p, b\}, \{q\}).$$

Intuitively, this reaction requires p and b together to produce q. All entities other than p and b inhibit this reaction, since such a situation would indicate an impossible condition for a *deterministic* finite transition system. Altogether, the reactions of $\mathcal{RS}_{\mathbf{F}}$ are exactly these transition-based reactions:

$$A_{\mathbf{F}} = \{a_{p,b,q} \mid \delta(p, b) = q\}.$$

As an example, suppose that $\mathbf{F}$ is the following three-state transition system.
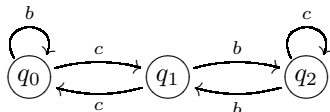
Then $S_{\mathbf{F}} = \{q_0, q_1, q_2, b, c\}$ and $A_{\mathbf{F}}$ contains exactly these six reactions:

$(\{q_0, b\}, \{q_1, q_2, c\}, \{q_0\})$, $\quad (\{q_0, c\}, \{q_1, q_2, b\}, \{q_1\})$, $\quad (\{q_1, b\}, \{q_0, q_2, c\}, \{q_2\})$,

$(\{q_1, c\}, \{q_0, q_2, b\}, \{q_0\})$, $\quad (\{q_2, b\}, \{q_0, q_1, c\}, \{q_1\})$, $\quad (\{q_2, c\}, \{q_0, q_1, b\}, \{q_2\})$.

In the transition system, we can consider this transition sequence:

$$q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_2 \xrightarrow{c} q_2 \xrightarrow{b} q_1 \xrightarrow{c} q_0$$

which corresponds to this interactive process in $\mathcal{RS}_{\mathbf{F}}$:
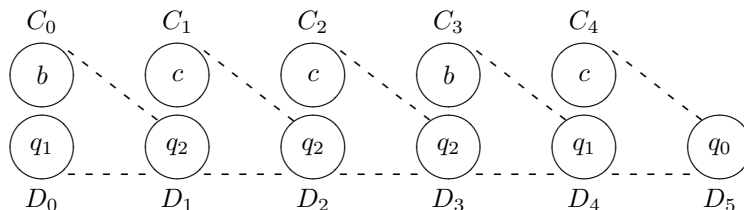

If the transition system has only one state and one input character, then our construction creates reactions with no inhibitors, violating Definition 1. This can be fixed by adding an artificial entity to the background set for the purpose of being the lone inhibitor in all reactions. The construction also fails if Q and Σ are not disjoint, since Q and Σ must be disjoint parts of the background set in the construction.

This paper is of a tutorial nature, so from this standpoint (and because of space limitations), the deterministic aspect of the previous example is didactically suitable here. But natural modifications to the construction yield the standard as well as novel nondeterministic models. For example, we could omit the states from the inhibitor sets of the reactions. This would allow nondeterminism via several possible mechanisms such as multiple start states or the usual notion of a transition relation (rather than a function). Similarly, we can determine which characters may "occur concurrently" and which may not.

5. Combinatorics of Reaction Systems

When we consider only context-independent processes of a reaction system $\mathcal{A} = (S, A)$, then the state transition function is simply the result function, $res_{\mathcal{A}} : 2^S \to 2^S$. In this case the current state completely determines (through $res_{\mathcal{A}}$) the successor state. Such a transition function may be considered as a total function (where the empty set is a possible result) or, as a variation, we may consider it to be a partial function that is undefined at any point $T \subseteq S$ such that no

$a \in A$ is enabled by T. Hence one can consider reaction systems as specifications of finite functions.

This functional view of reaction systems leads to questions such as how restrictions on reactions translate into restrictions on the ensuing result functions (see [8]). One initial result in this direction entails the requirement that each individual reaction has no more than a fixed number of *resources.* (Recall that the resources are the combination of the reactants and the inhibitors.) In particular, a reaction system (S, A) is an (s, m) reaction system provided that $|S| = s$ and each reaction $a \in A$ has no more than m resources (i.e., $m_a \leq m$). From this definition, the family of functions $\mathcal{F}_{s,m}$ is then defined as the set of functions that are isomorphic to the result functions of (s, m) reaction systems (we use here the definition of function isomorphism from our Preliminaries).

Definition 10. *For integers $s \geq m \geq 2$, the family $\mathcal{F}_{s,m}$ of functions is defined as*

$\{f \mid there\ exists\ an\ (s, m)\ reaction\ system\ \mathcal{A}\ such\ that\ f\ is\ isomorphic\ to\ \mathrm{res}_{\mathcal{A}}\}$

It turns out that, for a given $s \geq 2$, increasing resources expands the family of definable functions.

Theorem 11 ([8]) *For all integers $s \geq 2$: $\mathcal{F}_{s,2} \subsetneq \mathcal{F}_{s,3} \subsetneq \cdots \subsetneq \mathcal{F}_{s,s}$.*

This result is derived in terms of result functions that are partial functions, but an equivalent result occurs for total result functions.

At the lowest point of this hierarchy, $\mathcal{F}_{s,2}$, all of the reactions have only a single reactant and a single inhibitor, yet even there the functions can exhibit elaborate iterative behavior. To understand this behavior, let $T \subseteq S$ be a subset of the background entities and consider the effect of repeatedly applying a result function $res_{\mathcal{A}}$ to generate the sequence T, $res_{\mathcal{A}}(T)$, $res_{\mathcal{A}}(res_{\mathcal{A}}(T))$, $\cdots$. Because there are finitely many states, any such sequence must ultimately become cyclic, as shown in Figure 2 (using $\Rightarrow$ to represent each application of $res_{\mathcal{A}}$ and using $res_{\mathcal{A}}^i$ for the i-fold composition of $res_{\mathcal{A}}$ with itself).

$$T \Rightarrow res_{\mathcal{A}}(T) \Rightarrow \mathrm{res}_{\mathcal{A}}^2(T) \Rrightarrow$$

Fig. 2. Cyclic sequence in a reaction system

Such a sequence has two parts: the state transitions that occur before entering the cycle, called the *lead-in* (of length j in Figure 2), and the transitions within the cycle itself (of length k in Figure 2). In some sequences, the cycle is trivial, consisting only of the empty set that repeatedly maps to itself (in which case, we say that such a sequence, and hence the system *has died*). In general, though, both the lead-in and cycle parts of a sequence can be long, and this is what we mean by elaborate behavior. In fact, even for an $(s, 2)$ reaction system, the length of each part of the sequence can be exponential in the size of S:

Theorem 12 ([8]) *Let $k \geq 1$ be an integer. Then there exists a $(3k, 2)$ reaction system with a $(3 \cdot 2^k - 3)$-step state sequence that dies; and there exists a $(3(k+1), 2)$ reaction system with a $(3 \cdot 2^k - 1)$-step cycle.*

Detailed combinatorial properties of reaction system functions can be analyzed by placing more exacting restrictions on the number and kind of reactions (see [7]). For example, the following theorem gives a formula for the probability that a random reaction is enabled. (A reaction (R, I, P) with $|R| = r$, $|I| = i$ and $|P| = p$ is referred to as a *reaction of type* (r, i, p); also *random* means that the reaction is selected with a uniform probability distribution from the set of all possible reactions of the given kind.)

Theorem 13 ([7]) *Consider a background set of size s and a state T that contains a certain proportion, μs, of entities from the background set. For positive integers r, i, and p (with $r + i \leq n$ and $p \leq n$), the probability that a random (r, i, p) reaction is enabled by T is given by the following closed formula using binomial coefficients:*

$$\frac{\dbinom{\mu s}{r} \times \dbinom{(1 - \mu)s}{i}}{\dbinom{s}{r} \times \dbinom{s - r}{i}} .$$

For large s, this value approaches the limit $\mu^r (1 - \mu)^i$. The formula can be used to predict the probability of a state transition from any given size to any other given size, and these results have been confirmed via Monte Carlo experiments run at the University of Colorado [7]. These experiments are particularly interesting in that they study discrete transition systems that are exponentially large in the size of the background set—and yet, they are manageable and analyzable because of the structure provided by reaction systems.

The limit formula, $\mu^r (1 - \mu)^i$, has also been used as a basis for "designer reaction systems"—systems that are specifically designed so that the state sequences of the result function have very little probability of dying. This is done by selecting the number and kind of reactions so that the expected size of the next state is quite close to half the size of the background set. In this way, the state rarely gets too small (causing death by lack of reactants) or too large (death by too much inhibition). For example, random systems can be created in which the number of reactions is a

multiple of the size of the background set. With $5.5s$ reactions of type (1,3,1) and $5.5s$ reactions of type (3,1,1), the limit formula predicts that a system will rarely die; and, in fact, for a variety of systems with $s \geq 28$, the Monte Carlo trials never found a death.

The low probability of death is similar to the expectation for completely random functions on exponentially large sets, but other behavior differs. For example, the expected cycle length for a totally random function on a domain of size 2^s is given by a function that Knuth [9] called the Q-function:

$$Q(2^s) = \sqrt{\frac{\pi}{2}}(2^s)^{0.5} - \frac{1}{3} + \frac{1}{12}\sqrt{\frac{\pi}{2}}(2^s)^{-0.5} - \frac{4}{135}(2^s)^{-1} + \frac{1}{288}\sqrt{\frac{\pi}{2}}(2^s)^{-1.5} + O((2^s)^{-2}).$$

Our experimental results (see [7]) show that the $5.5s$ reaction systems have average cycle lengths that approach $Q(2^{0.74s})$.

6. Reaction Systems with Duration

Recall that there is no permanency in reaction systems—an entity is not retained in a transition of an interactive process unless it is either produced by some reaction or introduced through context. On the other hand, it is well known that a decay of entities in a biochemical environment (e.g., macromolecules in a living cell) requires some time to be realized (the decay time). To take this into account one considers reaction systems with duration [1]—here an entity x produced in a state W_{i+1} of an interactive process $W_0, \cdots, W_{i+1}, \cdots, W_m$ with $0 \leq i \leq m - 1$ will live (will be present/retained) at least in states $W_{i+1}, \cdots, W_{i+d(x)}$, where $d(x)$ is the duration of x and $m \geq i + d(x)$. In the above, "produced in W_{i+1}" means that $x \in \mathrm{res}_{\mathcal{A}}(W_i)$, where $\mathcal{A}$ is a reaction system with duration. In fact x can live beyond $W_{i+d(x)}$ if either x is introduced by context or x is also produced in one of the states $W_{i+2}, \cdots, W_{i+d(x)}$. Formally, we get the following definition.

Definition 14. *A reaction system with duration, abbreviated rsd, is a triplet $\mathcal{A} = (S, A, d)$, where (S, A) is a rs, and $d : S \to \mathbb{Z}^+$.*

We refer to (S, A) as the *underlying rs of $\mathcal{A}$*, denoted by $\mathrm{und}(\mathcal{A})$, and to d as the *duration function of $\mathcal{A}$*. For $x \in S$, $d(x)$ is the *duration of x* (in $\mathcal{A}$). Intuitively, once an element x is produced, its lifetime is $d(x)$ consecutive states.

The notion of interactive process is extended now to incorporate duration according to the above intuition. A graphical illustration of an interactive process is provided in Figure 3.

Definition 15. *Let $\mathcal{A} = (S, A, d)$ be a rsd. An* interactive process *in $\mathcal{A}$ is a triplet $\pi = (\gamma, \delta, \rho)$ of finite sequences such that, for some $n > 1$, $\gamma = C_0, C_1, \cdots, C_n$, $\delta = D_0, \cdots, D_n$, $\rho = G_0, \cdots, G_n$ where for all $i \in \{0, \cdots, n\}$, $C_i, D_i, G_i \subseteq S$,*

$$G_i = \{x \in S \mid d(x) \geq 2 \text{ and } x \in D_j, \text{for some } j \in \{i - (d(x) - 1), \cdots, i - 1\}\},$$

$D_0 = \varnothing$, and $D_i = \mathrm{res}_{\mathcal{A}}(D_{i-1} \cup C_{i-1} \cup G_{i-1})$ *for $i > 0$.*

As before, our intuition is that the C_i are the new context entities entering the system, and the D_i are the immediate results of the system's reactions. However the G_i are entities that persist after being produced in a previous state. For a given interactive process π in a rsd $\mathcal{A}$, the sequence $\tau = W_0, W_1, \cdots, W_n$ defined by $W_i = C_i \cup D_i \cup G_i$ for all $i \in \{0, \cdots, n\}$ is the *state sequence of* π. The notions and notations of context-independent state sequence, $STS(\mathcal{A})$, and $CISTS(\mathcal{A})$ for $\mathcal{A}$ carry over from reaction systems through the above notion of an interactive process.

Note that if $d(x) = 1$ for each $x \in S$, then one gets "ordinary" reaction systems. In this case, the rsd $\mathcal{A} = (S, A, d)$ is "essentially" the rs $(S, A) = und(\mathcal{A})$, meaning that the result and state sequences are *not* influenced by the duration sequence $G_0 = \varnothing, G_1 = \varnothing, \cdots, G_n = \varnothing$. Thus, $D_0 = \varnothing$ and for all $i \in \{1, \cdots, n\}$, $D_i = res_{\mathcal{A}}(D_{i-1} \cup C_{i-1})$ and $W_i = C_i \cup D_i$.

Example 16. *Consider again the reaction system* $\mathcal{U} = (S, U)$ *with* $S = \{1, 2, 3, 4\}$ *from Example 7. We have seen that* $\tau = \{1, 2, 4\}, \{1, 2, 3\}, \varnothing$ *is a context-independent state sequence in* $\mathcal{U}$. *Hence* τ *"dies" in the third state.*

Define now $d : S \to \mathbb{Z}^+$ *by* $d(1) = 2$, $d(2) = d(3) = 1$ *and* $d(4) = 3$, *and consider the rsd* $\mathcal{U}_d = (S, U, d)$. *Let us begin again with* $W_0 = \{1, 2, 4\}$. *We get then the following context-independent state sequence* τ' *of* $\mathcal{U}_d$: $W_0 = \{1, 2, 4\}$, $W_1 = \{1, 2, 3\}$, $W_2 = \{1\}$, $W_3 = \{2\}$, $W_4 = \{1, 3\}$, $W_5 = \{1, 2, 3\}$, $W_6 = \varnothing$. *Here, e.g., in the notation of Definition 15 we have* $W_2 = D_2 \cup G_2 = \varnothing \cup \{1\}$. *Although we begin in the same initial state, we get a different "dying sequence" in* $\mathcal{U}_d$ *(compared to* $\mathcal{U}$).

It turns out that duration of entities in a rsd can be explained through interactions with an environment. More precisely, given $\mathcal{A} = (S, A, d)$ one can embed $und(\mathcal{A})$ in a rs (without duration) $\mathcal{A}' = (S', A')$, where embedding means $S \subseteq S'$ and $A \subseteq A'$ (and we write then $und(\mathcal{A}) \sqsubseteq \mathcal{A}'$), so that the set of context-independent state sequences of $\mathcal{A}$ is obtained by projection onto S of context-independent state sequences of the bigger system $\mathcal{A}'$. In this way, the phrase "embedding a rs in an environment" becomes a formal term/construct, and indeed the duration is the result of an interaction of a rs with its environment. This is formally stated by the following result.
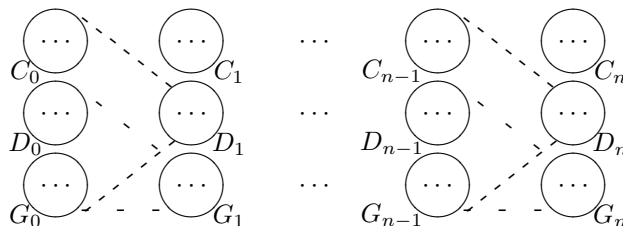

Fig. 3. Interactive process for a reaction system with duration.

14   *Brijder, Ehrenfeucht, Main, Rozenberg*

Theorem 17 ([1]) *For every rsd system $\mathcal{A}$ there exists a rs $\mathcal{A}'$ such that* $\mathrm{und}(\mathcal{A}) \sqsubseteq \mathcal{A}'$, *and* $\mathrm{proj}_S(CISTS(\mathcal{A}')) = CISTS(\mathcal{A})$.

Clearly, if $\mathcal{A}$ is a reaction system (without duration) and $W_0, \cdots, W_i, W_{i+1}, \cdots, W_j, W_{j+1}, \cdots, W_m$ is a context-independent state sequence of $\mathcal{A}$ such that $W_i = W_j$, then also $W_{i+1} = W_{j+1}$. This follows from the fact that such a state sequence results from iterating the function $res_{\mathcal{A}}$. However, this is not the case in reaction systems with duration, as was illustrated in Example 16, where $W_1 = W_5$ but $W_2 \neq W_6$. In particular, in reaction systems one may have a state sequence $W_0, \cdots, W_i, W_{i+1}, W_{i+2}, \cdots, W_m$ such that $W_i = W_{i+1}$ but $W_{i+1} \neq W_{i+2}$ as illustrated in the following example.

Example 18. *Consider the rsd $\mathcal{V} = (\{1,2\}, A, d)$ with $A = \{(\{2\}, \{1\}, \{1,2\})\}$ and $d(1) = 3$ and $d(2) = 2$. We have that $\tau = W_0, W_1, W_2, W_3 = \{2\}, \{1,2\}, \{1,2\}, \{1\}$ is a context-independent state sequence of $\mathcal{V}$. Notice that there are two consecutive states (W_1, W_2) which are equal while the next state (W_3) is again different.*

Such "locally constant" behavior of context-independent state sequences is formally defined as follows.

Definition 19. *Let $\mathcal{A}$ be a rsd, and let $\tau = W_0, W_1, \cdots, W_n \in STS(\mathcal{A})$. For $i \in \{1, \cdots, n\}$, W_i is* locally constant *if $W_i = W_{i-1}$.*

Locally constant behavior leads to blocks/segments consisting of the same state. We will discuss some of their properties now.

We first need some formal notions and notation. Let $\tau = W_0, W_1, \cdots, W_n$ be a sequence of sets. A *segment τ' of τ* is a consecutive subsequence $W_i, W_{i+1}, \cdots, W_j$ of τ for some $i, j \in \{0, \cdots, n\}$ with $i \leq j$. We say that τ' is *nonterminal in τ* if $j < n$. Also, τ' is *constant* if $W_i = W_{i+1} = \cdots = W_j$, and it is *maximally constant in τ* if it is constant and it cannot be extended to a longer constant segment of τ (i.e., either $i = 0$ or $W_{i-1} \neq W_i$, and either $j = n$ or $W_j \neq W_{j+1}$).

It turns out that long-enough context-independent state sequences must contain locally constant segments.

Theorem 20 ([1]) *Let $\mathcal{A} = (S, A, d)$ be a rsd, and let $t_0 = \min\{d(x) \mid x \in S\}$. Let $\tau = W_0, W_1, \cdots, W_n \in CISTS(\mathcal{A})$, and let $\rho = W_i, W_{i+1}, \cdots, W_{i+t_0-1}$ be a segment of τ such that $i \geq 1$. If $t_0 > 2|S|$, then there exists a state in ρ which is locally constant.*

Using Theorem 20 one can derive a lower bound on the average length of maximal constant segments.

Corollary 21. *Let $\mathcal{A} = (S, A, d)$ be a rsd, and let $t_0 = \min\{d(x) \mid x \in S\}$. If $t_0 \geq 2|S|$, then the average length of maximal constant segments in context-independent state sequences of $\mathcal{A}$ longer than t_0 is at least $\frac{t_0 - 2|S|}{2|S|+1}$.*

Finally, we have the following upper bound on maximal constant segments.

Theorem 22 ([1]) *Let $\mathcal{A} = (S, A, d)$ be a rsd, and let $t_1 = \max\{d(x) \mid x \in S\}$. For every $\tau \in CISTS(\mathcal{A})$, if σ is a nonterminal maximal constant segment of τ, then $|\sigma| \leq t_1$.*

7. Causalities in Reaction Systems

Let us recall that a reaction system is a formal model of interactions between biochemical reactions where the underlying mechanisms of these interactions are facilitation and inhibition. Hence the understanding of causalities in reaction systems (i.e., the ways that the entities influence each other) is important for the understanding of the functioning of reaction systems. Some of these causalities are "directly visible" from the specification of a rs, e.g., for entities x and y there is a production b such that $y \in M_b$ and $x \in P_b$—then the production of x via b is dependent on the presence or absence of y (and, perhaps, on other entities). This type of causality is informally referred to as a *static* causality. Another type of causality, informally referred to as *dynamic*, is defined as a property of the set of all state sequences of a given rs. In this section we will consider both kinds of causalities.

Recall that for a reaction a, we define the resources of a as $M_a = R_a \cup I_a$. Given a reaction a, we will refer to "forward causalities" (from the resources M_a to the products P_a) as *influence*, and to the "backward/inverse causalities" (from the product P_a to the resources M_a) as *dependence*. Formally, both static causalities, the influence and the dependence, are defined as follows.

Definition 23. *Let $\mathcal{A} = (S, A)$ be a rs, and let $x \in S$. The* resource dependence set of x *(denoted by* MD_x*), and the* product influence set of x *(denoted by* PI_x*) are defined by:*

$$\mathrm{MD}_x = \bigcup_{a \in A, x \in P_a} M_a \qquad \text{and} \qquad \mathrm{PI}_x = \bigcup_{a \in A, x \in M_a} P_a \ .$$

Now that the resource dependence and product influence are defined for entities, we can formally define reaction systems where these kinds of causalities are bounded.

Definition 24. *Let $\mathcal{A} = (S, A)$ be a rs and let $q \in \mathbb{Z}^+$.*

- *$\mathcal{A}$ is a rs with q-bounded resource dependence, abbreviated by q-MD rs, if $|\mathrm{MD}_x| \leq q$ for each $x \in S$.*
- *$\mathcal{A}$ is a rs with q-bounded product influence, abbreviated by q-PI rs, if $|\mathrm{PI}_x| \leq q$ for each $x \in S$.*
- *The* average resource dependence of $\mathcal{A}$*, denoted by* $\mathrm{avMD}(\mathcal{A})$*, is defined as $\sum_{x \in S} \frac{|\mathrm{MD}_x|}{|S|}$.*
- *The* average product influence of $\mathcal{A}$*, denoted by* $\mathrm{avPI}(\mathcal{A})$*, is defined as $\sum_{x \in S} \frac{|\mathrm{PI}_x|}{|S|}$.*

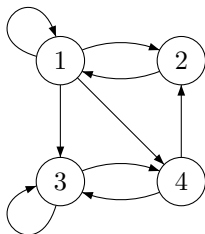16 *Brijder, Ehrenfeucht, Main, Rozenberg*

Fig. 4. The influence graph from Example 27.

Example 25. *Let $\mathcal{A} = (S, A)$ with $S = \{1, 2, 3, 4\}$ and $A = \{(\{1\}, \{4\}, \{2\}),$ $(\{2\}, \{1\}, \{1\}), (\{3\}, \{1, 4\}, \{3\}), (\{1\}, \{3\}, \{4\})\}$. We have $\mathrm{MD}_1 = \{1, 2\}$, $\mathrm{PI}_1 = \{1, 2, 3, 4\}$, $\mathrm{MD}_2 = \{1, 4\}$, $\mathrm{PI}_2 = \{1\}$, $\mathrm{MD}_3 = \{1, 3, 4\}$, $\mathrm{PI}_3 = \{3, 4\}$, $\mathrm{MD}_4 = \{1, 3\}$, and $\mathrm{PI}_4 = \{2, 3\}$. Thus $\mathcal{A}$ is a 3-MD and a 4-PI rs. Moreover, $\mathrm{avMD}(\mathcal{A}) = \mathrm{avPI}(\mathcal{A}) = 9/4$.*

The static causalities in a rs $\mathcal{A}$ are directly depicted in the influence graph of $\mathcal{A}$ which is formally defined as follows.

Definition 26. *Let $\mathcal{A} = (S, A)$ be a rs. The influence graph of $\mathcal{A}$, denoted by $\inf_{\mathcal{A}}$, is the digraph (S, E), where for $x, y \in S$, $(x, y) \in E$ if and only if $x \in M_a$ and $y \in P_a$ for some $a \in A$.*

Example 27. *Consider again the reaction system $\mathcal{A}$ from Example 25. The influence graph of $\mathcal{A}$ is depicted in Figure 4.*

Influence graphs are useful in analysis of causalities in reaction systems. For example, they yield a simple proof of the following property.

Theorem 28 ([2]) *For every rs $\mathcal{A}$, $\mathrm{avMD}(\mathcal{A}) = \mathrm{avPI}(\mathcal{A})$.*

We move now to consider dynamic causalities (ω denotes the smallest infinite ordinal, which is bigger than any natural number). The key notion here is the notion of the causal distance from entity x to entity y which formalizes the "minimal influence distance of x on y" within all possible context-independent state sequences. The intuition behind this notion is as follows. If we consider a pair $\tau = W_0, \cdots, W_m$ and $\tau' = W_0', W_1', \cdots, W_m'$ of two context-independent state sequences such that $W_0 \oplus W_0' = \{x\}$ and we compare pairwise consecutive states of τ and τ', then if, for some $j \in \{1, \cdots, m\}$, $y \in W_j \oplus W_j'$ then this appearance of y in one of W_j, W_j' is caused by x. If such a j does not exist, then x does not influence y within the pair τ, τ', and we set the causal distance equal to ω. If, on the other hand, n is the minimal such j, then n is the causal distance from x to y within this pair τ, τ'. Now consider the causal distance from x to y in the family of all pairs of context-independent state sequences where the symmetric difference of initial states equals x, then we get the causal distance from x to y (in the given rs $\mathcal{A}$).

The notion of causal distance formalizes the minimal "influence distance" between two entities x and y within all possible context-independent state sequences.

Definition 29. *Let $\mathcal{A} = (S, A)$ be a rs, and $x, y \in S$.*

- *Let $\tau, \tau' \in CISTS(\mathcal{A})$ where $\tau = W_0, W_1, \cdots, W_m$, $\tau' = W_0', W_1', \cdots, W_m'$, and $W_0 \oplus W_0' = \{x\}$. Let moreover $Z_{x,y}(\tau, \tau') = \{n \in \{0, \cdots, m\} \mid y \in W_n \oplus W_n'\}$. Then the* causal distance *from x to y in τ, τ' is defined by:*

$$\delta_{x,y}(\tau, \tau') = \begin{cases} \min Z_{x,y}(\tau, \tau') & \text{if } Z_{x,y}(\tau, \tau') \neq \varnothing \\ \omega & \text{otherwise} \end{cases}.$$

- *The* causal distance *from x to y is defined by:*

$$\mathrm{cd}_{x,y} = \min\{\delta_{x,y}(\tau, \tau') \mid \tau, \tau' \in CISTS(\mathcal{A}), |\tau| = |\tau'|, \text{ and } \mathrm{init}(\tau) \oplus \mathrm{init}(\tau') = \{x\}\}.$$

Example 30. *For the reaction system $\mathcal{A}$ from Example 25, $\tau_1 = \{2\}, \{1\}, \{2, 4\}$ and $\tau_2 = \{1, 2\}, \{2, 4\}, \{1\}$ are two context-independent state sequences of $\mathcal{A}$. Note that $W_0 \oplus W_0' = \{1\}$. We have $\delta_{1,1}(\tau, \tau') = 0$, $\delta_{1,2}(\tau, \tau') = \delta_{1,4}(\tau, \tau') = 1$, and $\delta_{1,3}(\tau, \tau') = \omega$.*

It turns out that, for a q-PI reaction systems $\mathcal{A}$, the number of entities with causal distance d from any entity x is at most q^d.

Theorem 31 ([2]) *Let $\mathcal{A} = (S, A)$ be a rs and let $x \in S$. If $\mathcal{A}$ is a q-PI rs for $q \geq 1$, then for every $d \in \mathbb{Z}^+$, $|\{y \in S \mid \mathrm{cd}_{x,y} = d\}| \leq q^d$.*

8. Discussion

In this paper, we have introduced reaction systems along with their underlying intuition and motivation. The two primary examples—a binary counter and finite transition systems—are familiar to our intended audience of theoretical computer scientists. Finally, we gave a guided tour of results from three theoretical research areas within reaction systems.

Because of the restriction on the size of this paper, the tour was short. In order to give a more complete picture of (research in) reaction systems we will briefly discuss now some other research themes from the literature.

Although the research on reaction systems began with the investigation of a specific formal model, by today we have a whole *framework* for an abstract study of biochemical reactions based on reaction systems. Within this framework the basic notion of a reaction system can be modified whenever needed in studies of various research themes concerning biochemical reactions. Reaction systems with duration from Section 6 are an example of such a modification—a reaction system was equipped with a duration function.

There is no counting in reaction systems, however there are many situations in biochemical systems where one needs to assign quantitative parameters to the states of a system. Our assumption is that a numerical value can be assigned to a

state T if there is a measurement of T yielding this value. To formalize this, one defines *measurement functions* which assign reals to states. Measurement functions are required to be additive, i.e., for a measurement function f and disjoint sets W_1, W_2 we have $f(W_1 \cup W_2) = f(W_1) + f(W_2)$. Consequently, a *reaction system with measurements* [5] is a reaction system equipped with a finite set of measurement functions, hence a triplet $\mathcal{A} = (S, A, F)$, where (S, A) is a rs and F is a finite set of measurement functions on (the subsets of) S.

An important case when one needs measurement functions is assigning time moments to states. This leads to fundamental questions such as "What is time in (models of) biochemical systems?" and "How can one capture the notion of time in the framework of reaction systems?". The problem of introducing time in reaction systems is investigated in [5]. Here a time function is a measurement function which has nonnegative values (no entity can set time back) and is such that if W, U are two consecutive elements of a state sequence, with U following W, then $f(W) < f(U)$ (time flows forward). Hence time is physical/material—it is embedded in (carried by) the entities of a system. It is shown in [5] how defining time in this way allows one to investigate notions such as reaction times, the time distances between states in a state sequence, and all kinds of rates.

Another important issue is the formation of structures in biochemical systems (biological organisms), see, e.g., [11]. This issue is investigated in [4], where it is demonstrated that dynamic runs of reaction systems (interactive processes) lead to the formation of "compounds" (modules), and that the family of all modules in a given state of a reaction system forms a lattice. Hence (as biochemists and biologists know well) the development of a biochemical system (an organism) leads to the formation of structures. It is also shown in [4] that reaction systems can be considered as self-organizing systems.

Acknowledgements

References

[1] R. Brijder, A. Ehrenfeucht, and G. Rozenberg. Reaction systems with duration, to appear, (2010).

[2] R. Brijder, A. Ehrenfeucht, and G. Rozenberg. A Note on Causalities in Reaction Systems, *Electronic Communications of the EASST* **30** (2010).

[3] A. Ehrenfeucht and G. Rozenberg. Reaction systems, *Fundamenta Informaticae* **76** (2006) 1-18.

[4] A. Ehrenfeucht and G. Rozenberg. Events and modules in reaction systems, *Theoretical Computer Science* **376** (2007) 3-16.

[5] A. Ehrenfeucht and G. Rozenberg. Introducing time in reaction systems, *Theoretical Computer Science* **410** (2009) 310-322.

[6] A. Ehrenfeucht and G. Rozenberg. Reaction Systems: A Model of Computation Inspired by Biochemistry, *Lecture Notes in Computer Science* **6224** (2010) 1-3.

[7] A. Ehrenfeucht, M. Main and G. Rozenberg. Combinatorics of life and death for reaction systems, *International Journal of Foundations of Computer Science* **22** (2009) 345-356.

[8] A. Ehrenfeucht, M. Main and G. Rozenberg. Functions defined by reaction systems, *International Journal of Foundations of Computer Science* to appear.

[9] D. E. Knuth. *The Art of Computer Programming* (Volume 1), *Fundamental Algorithms* (Addison-Wesley, Reading, 1968).

[10] G. Rozenberg and J. Engelfriet. Elementary Net Systems. *Lecture Notes in Computer Science* **112** (1998) 12-121.

[11] G. Schlosser and G.P. Wagner, eds. *Modularity in Development and Evolution*, The University of Chicago Press, Chicago and London, 2004.