DOI: 10.1002/chem.200801850

Systems Chemistry: Logic Gates, Arithmetic Units, and Network Motifs in Small Networks

Nathaniel Wagner and Gonen Ashkenasy*[a]

Abstract: A mixture of molecules can be regarded as a network if all the molecular components participate in some kind of interaction with other molecules—either physical or functional interactions. Template-assisted ligation reactions that direct replication processes can serve as the functional elements that connect two members of a chemical network. In such a process, the template does not necessarily catalyze its own formation, but rather the formation of another molecule, which in turn can operate as a template for reactions within the network medium. It was postulated that even networks made up of small numbers of molecules possess a wealth of molecular information sufficient to perform rather complex behavior. To probe this assumption, we have constructed virtual arrays consisting of three replicating molecules, in which dimer templates are capable of catalyzing reactants to form additional templates. By using realistic parameters from peptides or DNA replication experiments, we simulate the construction of various functional motifs within the networks. Specifically, we have designed and implemented each of the three-element Boolean logic gates, and show how these networks are assembled from four basic "building blocks". We also show how the catalytic pathways can be wired together to perform more complex arithmetic units and network motifs, such as the half adder and half subtractor computational modules, and the coherent feed-forward loop network motifs under different sets of parameters. As in previous studies of chemical networks, some of the systems described display behavior that would be difficult to predict without the numerical simulations. Furthermore, the simulations reveal trends and characteristics that should be useful as "recipes" for future design of experimental functional motifs and for potential integration into modular circuits and molecular computation devices.

Keywords: catalytic networks · computer chemistry · kinetics · logic gates · self-replication

Introduction

Rapid progress has been achieved during the last few years towards understanding network behavior in biological systems, due in part to the involvement of physicists, mathematicians, and computer scientists in this area of research. Much attention has been devoted to dissecting complex cellular networks into small functional modules that can be analyzed in detail. While one research direction focuses on the opportunity to artificially manipulate such modules, another direction is directed towards investigating the evolutionary basis for the formation of such modules.[1–7]

[a] Dr. N. Wagner, Dr. G. Ashkenasy
Department of Chemistry
Ben Gurion University of the Negev
Beer Sheva 84105 (Israel)
Fax: (+972) 8-647-2943
E-mail: gonenash@bgu.ac.il

Also recently, supramolecular chemistry has evolved to the level that it is now possible to artificially design small networks of dynamically interacting molecules. An important class of synthetic networks consists of the dynamic combinatorial libraries (DCLs), which are suitable for the search for new sensors and receptors.[8–14] Interestingly, during the last three years or so, chemists have started to probe the DCLs as models for systems behavior.[15,16] In this respect, they have studied how an individual library member can influence the behavior of all other members, or vice versa, namely, how the concentration of an individual library member is controlled by the properties of the other species in the mixture.

Systems that are related to the DCLs, while presenting some clear differences in dynamic behavior, may be constructed from small networks of replicating molecules. Over the years chemists have designed and characterized minimal self-replication systems made of nucleic acids (DNA[17–20] and RNA[21]), peptides,[22–25] mixed protein–nucleic acids,[26] and small organic molecules.[27–30] The catalytic principles of

all the molecular families have been utilized to devise networks that consist of two,[31–33] three,[34,35] four,[36–39] or larger numbers[40,41] of replicating molecules. The observed molecular replication within these networks is not only a consequence of autocatalysis, but also of cross-catalytic processes in which a template molecule is used to enhance the formation of different molecules, usually its own mutants.[42–46]

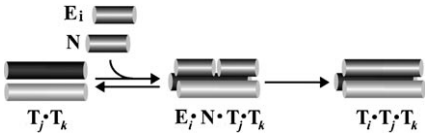
To further understand self-replicating systems and catalytic networks, their dynamics have been solved mathematically and computed numerically. An empirical rate equation was used early on to describe the time and concentration dependence of minimal self-replicators, solving for the reaction order p as a function of the system's equilibrium constants.[47] The experimental nucleotide-[17] and peptide-based[42] systems have been simulated and calibrated with experimental data, by modeling the systems and solving the differential equations using a Taylor series approximation.[48] The organic systems have been modeled kinetically and calculated numerically,[49] and a comprehensive kinetic model has succeeded in explaining the results under differing experimental conditions.[50,51] Single-template autocatalytic and cross-catalytic networks have been solved analytically under various steady-state assumptions and approximations, exploring the boundaries between competition and coexistence.[52] This approach was expanded to higher order replication reactions, leading to the observation of fixed points, bifurcations and "hypercycles".[53] By using the SimFit program, catalytic networks of nucleotides have been computed numerically by kinetic modeling and parameter fitting.[37] Experimental chiro-selective peptide networks[39] have been modeled and numerically solved, highlighting the heterochiral[54] and homochiral[55] cases.[56] Kinetic modeling of autocatalytic and cross-catalytic networks using a set of dimensionless differential equations has led to temporal and spatial patterns yielding bifurcations and oscillations.[57–59]

The DCLs and the non-enzymatic replicating networks can be considered as simple models for autonomous modules in cells. It was postulated that even networks made up of small numbers of molecules (e.g., ternary networks) possess a wealth of molecular information sufficient for exhibiting rather complex behavior. This phenomenon has different names, and was recently referred to as "systems chemistry".[29,60] In this paper we would like to probe theoretically and by simulation the following question: how much "systems chemistry" exists within simple small networks?

To answer this question, we have constructed virtual arrays consisting of three replicating molecules. Using realistic parameters from peptides or DNA replication experiments, we have simulated the existence of various different functional motifs within the networks. Specifically, we have shown how the networks should be manipulated in order to facilitate molecular replication through all Boolean logic operations, and how the catalytic pathways can be wired together to perform more complex computational motifs and feed-forward based network motifs. As in previous numerical studies of chemical networks, some of the systems described in this study display behavior that would be difficult to predict without numerical simulation. Furthermore, the simulations reveal fundamental trends and characteristics, which should be useful for future design of experimental functional motifs.

Results and Discussion

Replicating molecular networks: To investigate the kinetic behavior of small networks, we have constructed a collection of n molecules—conceptually representing nucleotides, peptides, or organic compounds—in which each molecule can serve as a template for its own formation and/or the formation of other molecule(s). The recurrent element in the network is a catalytic process that employs a dimeric template, which non-covalently binds to fragments of the products and enhances their ligation (Scheme 1). The dynamics of product formation in the system is a result of simultaneous progress of the various catalytic processes, which are in turn dictated by the competition for starting materials, productive or non-productive association of templates, and extent of product inhibition.



Scheme 1. General mechanism for auto- and cross-catalytic processes, in which template dimers of peptides, nucleotides, or organic compounds catalyze coupling of reactants forming trimer complexes, which can then dissociate into template dimers and free monomeric templates.

A general mechanism describing simultaneous auto and cross catalysis within our system can be drawn by Equations (1) and (2), in which $i, j, k = 1,\ldots n$.[47,52,53,58,59] In this description the dimers T_jT_k are the only catalytically active species, while there are given propensities for association of monomers to form the dimers and trimers. In addition to the catalytic reactions, each of the reactants E_i combines directly with N to form the templates T_i in a slower background reaction. We use the letters E and N for the reactants, since many systems are composed of electrophilic and nucleophilic fragments, but this is not required. Also, we pair n reactants of type E opposite only one reactant of type N, forming n templates of type T, but our analysis can easily be extended to include several reactants of type N.

$$\begin{aligned} &E_i + N + T_jT_k \underset{\langle a\rangle_{ijk}}{\overset{a}{\rightleftharpoons}} E_iNT_jT_k \xrightarrow{b_{ijk}} T_iT_jT_k \underset{\langle f\rangle}{\overset{f_{ijk}}{\rightleftharpoons}} \\ &T_i + T_jT_k \underset{\langle d\rangle}{\overset{d_{ijk}}{\rightleftharpoons}} T_i + T_j + T_k \end{aligned} \quad (1)$$

$$E_i + N \xrightarrow{g_i} T_i \quad (2)$$

In Equation (1), a, $\langle d\rangle$, and $\langle f\rangle$ are the diffusion-limited rate constants, assumed to be independent of i, j, and k; b_{ijk} are

the ligation rate constants, and are assumed to depend only on i, that is, $b_{ijk}=b_i$; d_{jk} are the dissociation constants of the intermediate dimers T_jT_k; and $\langle a\rangle_{ijk}$ and f_{ijk} are the dissociation constants of $E_iNT_jT_k$ and $T_iT_jT_k$, respectively. We note that d is symmetric, namely $d_{ij}=d_{ji}$, since T_iT_j is equivalent to T_jT_i. The dissociation constant f is also symmetric in the sense that $f_{ijk}=f_{jki}=f_{kij}$, since $T_iT_jT_k$, $T_jT_kT_i$, and $T_kT_iT_j$ are all equivalent. The constant f_{ijk} actually gives the total break-up rate of $T_iT_jT_k$, which, from symmetry, breaks up with equal probability into $T_i+T_jT_k$, $T_j+T_kT_i$, or $T_k+T_iT_j$. Similarly, $\langle a\rangle$ follows the sub-symmetry $\langle a\rangle_{ijk}=\langle a\rangle_{ikj}$. Apart from this, cross catalysis is not necessarily symmetric, since, for example, $E_iNT_jT_j$ is not the same as $E_jNT_iT_i$, and $T_iT_jT_j$ is not the same as $T_jT_iT_i$.

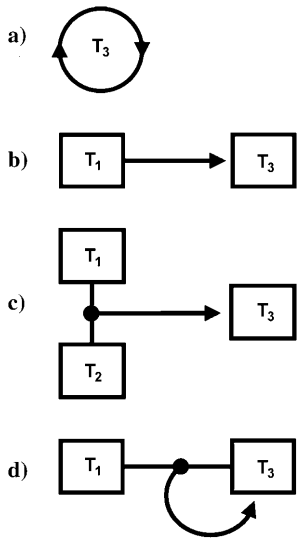
The g_i parameters in Equation (2) represent the rate constants of the background template-free ligation reactions. It should be noted that the background reactions affect the network in two aspects: 1) by producing templates and 2) by consuming starting material at the expense of the catalyzed reactions.

A system of several reactants and templates, with autocatalysis and mutual cross catalysis of various strengths, forms a molecular network. A productive network connection is evident by an efficient catalytic process leading to a template, while the interruption of a connection from one template to another corresponds to a very high value (greater by several orders of magnitude) for the specific dissociation constant $\langle a\rangle$. To verify the validity of our design, we first studied several extreme cases. For example, only autocatalytic pathways were allowed in one case, obtained by the assignment of high $\langle a\rangle_{ijk}$ to all pathways except those with $i=j=k$. In another case, all pathways were given the same catalytic efficiency resulting in a homogenous replication of the entire network (data for both cases not shown).

It was shown experimentally, by using the self-replicating systems, that all the kinetic constants, except for those considered diffusion limited, can be affected by choosing specific molecular design or experimental conditions. These experiments were conducted mainly to show how changing one or more of these parameters can improve the catalytic efficiency. The background reaction rate, for example, was slowed down to almost zero by choosing different chemical groups for pre-activation of the starting materials or by changing the reaction conditions (e.g., pH).[24,61] In other examples, the template effect exhibited by low $\langle a\rangle$ and f reaction constants was magnified by the design of optimal structural matching between two template molecules. Since structural matching also causes product inhibition (low f), which results in undesired parabolic replication, different approaches were taken to destabilize the template–product complex without compromising on stable template–reactant association.[32,62–64]

By systematically studying these systems through computer simulation, namely, by allowing or disallowing network connections that correspond to specific catalytic pathways, we can show how to actually synthesize logic gates, computational elements that are comprised of several gates, and network motifs for a variety of chemical systems.

Building blocks: By combining simple catalytic building blocks together, either in cooperation or competition, we can construct the more complex functional entities, such as logic gates, computational units or network motifs. In the above description, in which the T_jT_k dimers are catalysts for the formation of T_i, we consider four possible catalytic mechanisms in which $i=j=k$, $i\neq j=k$, $i\neq j\neq k$, or $i=j\neq k$. These mechanisms are essentially the following network catalytic building blocks (Scheme 2): 1) the autocatalytic building block, in which, for example, the dimer T_3T_3 catalyzes the formation of T_3 (Scheme 2a); 2) the homodimeric cross-catalytic block, in which, for example, the dimer T_1T_1 catalyzes the formation of T_3 (Scheme 2b); 3) the heterodimeric cross-catalytic block, in which, for example, the dimer T_1T_2 catalyzes the formation of T_3 (Scheme 2c); and 4) the combined auto-cross-catalytic building block, in which, for example, the dimer T_1T_3 catalyzes the formation of T_3 (Scheme 2d). We note that these building blocks are themselves functioning logic gates, namely that the autocatalytic and homodimeric cross-catalytic blocks are examples of IF (YES) gates, while the heterodimeric cross-catalytic and combined auto-cross-catalytic blocks are examples of AND gates.



Scheme 2. Graphical descriptions of the four basic catalytic building blocks: a) autocatalytic; b) homodimeric cross-catalytic; c) heterodimeric cross-catalytic; d) combined auto-cross-catalytic. More complex functional entities may be constructed from combining these blocks together, either in cooperation or competition.

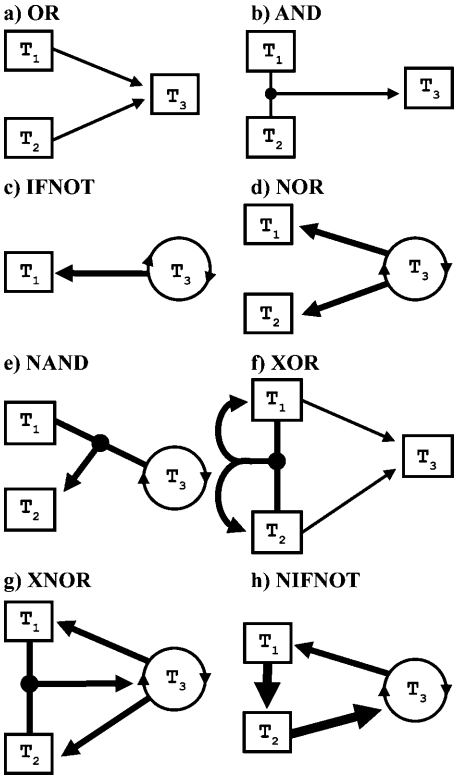
Logic gates: Boolean logic provides a simple and concise way to describe the output of processes that depend on more than one factor. Historically, the design and synthesis of chemical logic gates has started with the "bottom-up" ap-

proach, using molecules that were triggered chemically, electrochemically, or by light.[65–74] During the past years modules of biological systems have also been described by using Boolean logic, or have been intentionally manipulated to perform such operations.[35,69,75–90] We investigate here replicating molecular networks consisting of three templates [represented by $n=3$ in Eq. (1)], in which template-assisted product formation is controlled by all sixteen two-input logic operations.[91,92]

Table 1 shows truth tables for each of the (nontrivial) logic gates, along with the two basic one-input logic gates IF (YES) and NOT, and their symbols. The collection of net-

Table 1. Summary of the two one-input logic gates and the eight nontrivial two-input logic gates, showing their truth tables and symbols.

	0,0	1,0	0,1	1,1	Symbol
IF	0	1	–	–	–
NOT	1	0	–	–	–
OR	0	1	1	1	
AND	0	0	0	1	
IFNOT	0	1	0	0	
NOR	1	0	0	0	
NAND	1	1	1	0	
XOR	0	1	1	0	
XNOR	1	0	0	1	
NIFNOT	1	0	1	1	



Scheme 3. Graphical descriptions of the logic gates. An arrow from T_j to T_i depicts the cross-catalytic pathway in which the template dimer T_iT_j catalyzes the reactants E_i and N to form T_i. An arrow from the line joining T_j and T_k to T_i depicts the cross-catalytic pathway in which the template dimer T_jT_k catalyzes the reactants E_i and N to form T_i. Circular arrows depict autocatalytic pathways, and bold arrows represent stronger catalytic pathways.

work topologies in Scheme 3 shows the design of these logics with respect to the formation of T_3 molecules. It consists of the following eight examples: OR, AND, IFNOT (INHIBIT), NOR (Not OR), NAND (Not AND), XOR (eXcluded OR), XNOR (Not XOR or eXcluded NOR), and NIFNOT (Not IFNOT). We have designed the system such that an output of a certain combination of inputs will be taken as "1" if T_3 is formed through a template-assisted catalytic pathway, or as "0" if T_3 is formed slowly at a rate equal to or similar to the noncatalyzed background reaction.

Each of the logic gates can be constructed from the building blocks described above. Scheme 3 has been arranged to first include the simpler gates (OR and AND), and then the more complicated gates that require competition between, or integration of, more than one operation. It should be mentioned that some of the gates may also be formed within networks of other topologies, but those are not shown.

In the following paragraphs we describe the design and implementation of each of the simulated gates. First, each gate was constructed from the total network by allowing or interrupting the appropriate network connections, as described above. Then the exact values used for the rate constants were selected. For consistency, we used a default set of kinetic constants for the parameters shown in Equations (1) and (2) (see exact values in Table 2 and the Experimental Section). These defaults match experimental parameters obtained by fitting previous experiments to our current model,[35,41] and they correspond to realistic values from peptide or DNA replication experiments. When needed to perform specific gates, these values were modified conservatively, usually by up to an order of magnitude, in order to strengthen or weaken certain catalytic pathways. Additionally, we occasionally varied some of the parameters by slight factors for visual purposes, corresponding to "experimental conditions" for a given run. The exact values used for each case are specified in Table 2.

Each run used specific initial concentrations of reactants and templates (see also in Table 2 and the Experimental Section). E_3 and N were always initially present and not used as inputs. These two compounds and the other starting materials are essentially the gates' "hardware". The reactants E_1, E_2, E_3, and N were thus initialized each at 100 μM, except for the cases when E_1 or E_2 were used as inputs or were irrelevant. The templates T_1 and T_2 were initialized at zero, except when T_1 and/or T_2 were used as inputs. Template T_3, when not used as an input, was initialized at zero for the cases in which it was produced only cross-catalytically, and at 10 μM for the cases in which it was also produced autocatalytically.

Table 2. Summary of all gates, modules and motifs, listing the initial components, inputs, and parameters that differed from the default values.

	Initial components[a]	Inputs[a]	Parameter differing from defaults
OR	E_1, E_2 , E_3 , N	T_1, T_2	$\langle a\rangle_{322}=14.4$; $f_{322}=1200$
AND	E_1, E_2, E_3, N	T_1 , T_2	$b_1=8$; $g_1=0.8$
IFNOT	E_3, N	T_3 , E_1	$\langle a\rangle_{133}=0.1$; $b_i=1$; $f_{133}=100$; $g_i=0.1$
NOR	E_3, N, T_3	E_1, E_2	$\langle a\rangle_{133}=0.4$, $\langle a\rangle_{233}=0.1$; $b_i=1$; $f_{133}=200$, $f_{233}=100$; $g_i=0.1$
NAND	E_1, E_3, N, T_3	T_1, E_2	$\langle a\rangle_{213}=0.1$; $d_{33}=100$; $f_{213}=100$
XOR	E_1, E_2, E_3, N	T_1, T_2	$\langle a\rangle_{112}=\langle a\rangle_{212}=0.1$, $\langle a\rangle_{322}=10.5$; $b_i=1$; $d_{12}=1$, otherwise $d_{jk}=100$; $f_{112}=f_{212}=100$, $f_{322}=1025$; $g_i=0.1$
XNOR	E_3, N, T_3	E_1, E_2	$\langle a\rangle_{133}=\langle a\rangle_{233}=\langle a\rangle_{312}=0.1$; $b_i=1$; $d_{12}=d_{13}=1$, $d_{23}=2$; $f_{133}=f_{233}=f_{312}=100$; $g_i=0.1$
NIFNOT	E_3, N, T_3	E_1, E_2	$\langle a\rangle_{133}=\langle a\rangle_{233}=\langle a\rangle_{322}=\langle a\rangle_{211}=0.1$; $b_i=1$; $d_{11}=d_{22}=1$, $d_{33}=30$; $f_{133}=f_{233}=f_{322}=f_{211}=100$; $g_i=0.1$
computational module (Figure 2)	E_1, E_2, E_3, N	T_1, T_2	$\langle a\rangle_{111}=\langle a\rangle_{212}=\langle a\rangle_{322}=0.1$, $\langle a\rangle_{311}=0.025$; $b_i=0.1$; $d_{12}=1$, otherwise $d_{jk}=100$; $f_{111}=f_{212}=f_{322}=100$, $f_{311}=50$; $g_i=0.01$
network motif (Figure 3a)	E_1, E_2, E_3, N, T_1		$\langle a\rangle_{ijk}=111$
network motif (Figure 3b)	E_1, E_2, E_3, N, T_1		$\langle a\rangle_{ijk}=111$; $d_{12}=1000$, $d_{22}=1$
network motif (Figure 3c)	E_1, E_2, E_3, N, T_1		$\langle a\rangle_{ijk}=111$; $d_{11}=d_{22}=100$

[a] E_i, N = 100 μM; T_j = 10 μM.

For each gate we followed the production of T_3 for four possible combinations of two input variables. As specified in Figure 1 for each gate, the variables were either templates T_1, T_2, or T_3—for which a "1" input corresponds to 10 μM—or reactants E_1 or E_2—for which a "1" input corresponds to 100 μM.

The OR gate: The basic and simplest two-input logic gate is the OR gate (Scheme 3a). To perform this logic within the network, both T_1 and T_2 should catalyze the formation of T_3; that is, each of the dimers T_1T_1 and T_2T_2 can catalyze E_3 and N to form T_3. None of the two templates catalyze their own formation, and the back cross-catalytic reactions $T_3T_3\rightarrow T_1$ and $T_3T_3\rightarrow T_2$ are also not allowed. Hence, in computer simulation of the OR gate, initial high concentrations of either T_1 or T_2 are taken as the "1" inputs and enable the fast production of T_3 (Figure 1a). The simulation ran using the default reaction constants and initial conditions corresponding to realistic laboratory values (the default parameters are summarized in the Experimental Section). When no template is provided ("0,0" input), T_3 is produced by the slow background reaction. The "0,0" output is relatively negligible, but due to the weak background reaction, and then some catalysis once

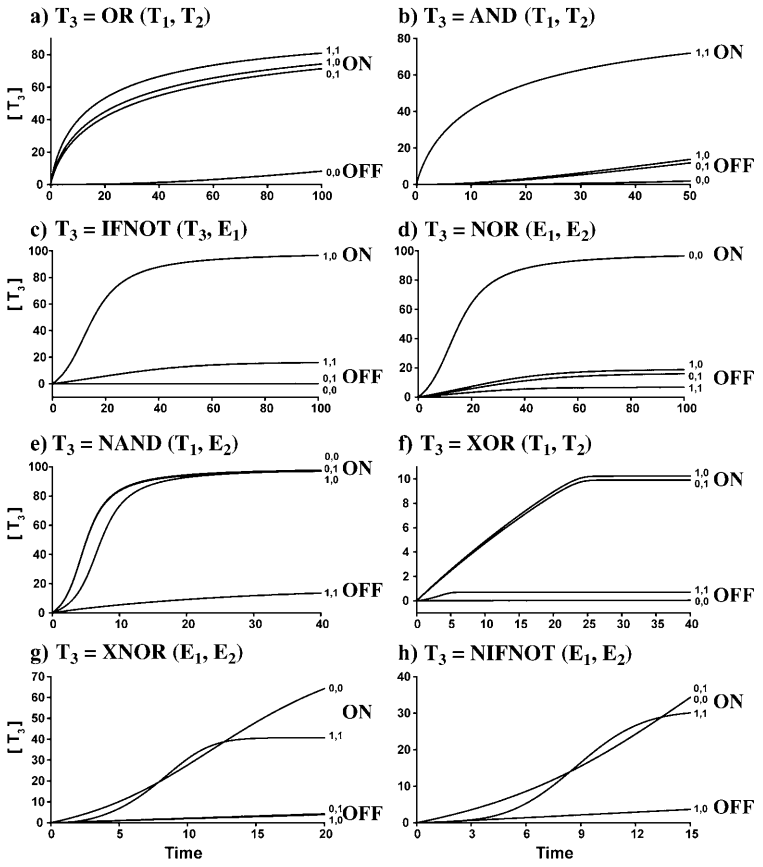


Figure 1. Dynamic simulations of the logic gates shown in Scheme 3. Each graph shows the production of T_3 (μM) as a function of time (min) for four possible combinations of two input variables. The exact parameters used and specific concentrations of reactants, templates and inputs are listed in Table 2 and the Experimental Section. In simulating the kinetics of some of the gates we took advantage of the "experimental conditions" as a means of distinguishing between the otherwise similar "1,0" and "0,1" cases (e.g., in the OR and XOR gates). The "0" output is relatively negligible in all cases, but due to the weak background reaction and some catalysis once small amounts of T are produced, not entirely zero.

small amounts of T_1 and T_2 are produced, not entirely zero. The "experimental" conditions (see also in the Experimental Section) led to slight differences between the "1,0" and "0,1" cases, while the "1,1" case is slightly stronger since both of these pathways are active in parallel.

The simulation results are similar to previously published experimental laboratory results.[35] We note that the studied OR gate is constructed dynamically and not structurally, that is, both T_1 and T_2 independently catalyze the formation of T_3 without any structural relationship between them. Alternatively, we could have also allowed the structural pathway, that is, the heterodimeric $T_1T_2 \rightarrow T_3$ cross catalysis, to contribute to the OR gate.

The AND gate: The AND gate is another simple three-element logic gate that is also a required participant in various complex logic devices, computational modules (e.g., half adder, full adder), and network motifs. The AND gate implemented here (Scheme 3b) consists of only one heterodimeric cross-catalytic building block, namely a building block that uses the T_1T_2 heterodimer for enhancing the formation of T_3. None of the other auto- or cross-catalytic pathways are allowed. We note that the AND gate design uniquely utilizes the properties of higher order catalysis, since the structural $T_1T_2 \rightarrow T_3$ pathway has no analogue in low order catalysis, in which only monomer templates act as catalysts.

Here too, the computer simulation ran using the default parameters and takes initial high concentrations of either T_1 or T_2 as the "1" inputs. Unlike the OR gate case, introducing only one of the templates in the "1,0" or "0,1" cases results in slow T_3 production. Only the "1,1" combination results in fast production of T_3 (Figure 1b). The non-zero OFF outputs are obtained due to weak background reactions and some limited catalysis that then follows. The slight differences between the "1,0" and "0,1" cases are due to "experimental" conditions, while the "0,0" case is even weaker, since the ensuing catalysis that follows, in the absence of initial concentrations of both T_1 and T_2, is negligible.

The IFNOT gate: The IFNOT gate, also known as INHIBIT logic, outputs a "1" when one input is "1" and the other input is "0". All other combinations yield a "0" output (see Table 1). This logic is a required component of subtractor computational modules. The dynamic IFNOT implemented here (Scheme 3c) follows the design of such a gate within peptide networks.[35] It is constructed from only two of the network elements, since the output T_3 is also used as an input for an autocatalytic building block. The inhibit logic behavior is observed as a result of the dynamics of an autocatalytic pathway $T_3T_3 \rightarrow T_3$ and a more efficient competing homodimeric cross-catalytic pathway $T_3T_3 \rightarrow T_1$. Specifically, the presence of T_3 enhances its own formation through autocatalysis, but when both T_3 and the electrophile E_1 are present, the cross-catalytic formation of T_1 will be dominant and T_3 will only be formed slowly.

T_3 and E_1 are thus the two inputs for simulation of the IFNOT gate (Figure 1c). For it to work properly, we ran the simulation with the default parameters for the autocatalytic pathway $T_3T_3 \rightarrow T_3$, while strengthening the cross-catalytic pathway $T_3T_3 \rightarrow T_1$. The latter was achieved by stabilizing the intermediate $E_1NT_3T_3$, namely by lowering the dissociation constant $\langle a \rangle_{133}$, even if the related dissociation constant f_{133} of $T_1T_3T_3$ decreased simultaneously. In addition, the most significant INHIBIT behavior was obtained when all of the background reaction constants (g_i) were kept low (see Experimental Section). As in the experimental IFNOT system, the "0,0" and "0,1" cases resulted in slow formation of T_3, the "1,0" input induced fast T_3 production, and the "1,1" case inhibited this autocatalysis significantly but not completely.

The NOR gate: The NOR logic is obtained from a combination of NOT and OR functions. In our design, the NOR gate consists of an autocatalytic building block $T_3T_3 \rightarrow T_3$ that competes dynamically with two more efficient homodimeric cross-catalytic blocks $T_3T_3 \rightarrow T_1$ and $T_3T_3 \rightarrow T_2$ (Scheme 3d). Accordingly, the gate "hardware" is made of E_3, N, and T_3, and its inputs are not templates but rather the reactants E_1 and E_2.

Computer simulation of the NOR gate yielded the graph shown in Figure 1d. As in the simulation of the IFNOT gate, we ran this simulation with default parameters for the T_3 autocatalysis, while strengthening the competing cross-catalytic pathways $T_3T_3 \rightarrow T_1$ and $T_3T_3 \rightarrow T_2$. This was achieved by stabilizing the intermediates $E_1NT_3T_3$ and $E_2NT_3T_3$, by lowering of the dissociation constants $\langle a \rangle_{133}$ and $\langle a \rangle_{233}$, respectively. Here again, the best results were obtained when we lowered the background reaction rate. As shown in Figure 1d, the fast autocatalysis of the "0,0" case is completely suppressed in the "1,1" case and also significantly inhibited in the "0,1" and "1,0" cases. The different kinetics in the last two cases are due to slightly different "experimental" conditions. These results are similar to previously published experimental laboratory results.[35]

The NAND gate: The NAND (Not AND) gate is of central importance in logic gate theory, since all possible logic operations may be constructed from combinations of NAND gates.[92,93] The gate's output is "1" when neither or either of the inputs are present, and "0" when both of the inputs are present (Table 1). In the NAND gate constructed here (Scheme 3e) T_3 is a good autocatalyst, while T_3 and T_1 are both required for a more efficient heterodimeric cross catalysis leading to the formation of T_2. The inputs are the initial concentrations of T_1 and E_2. If both T_1 and E_2 are present the cross-catalytic pathway will be dominant; otherwise the autocatalytic pathway dominates and produces T_3. The key to successful design of this NAND gate is to accelerate the competing cross-catalytic pathway $T_1T_3 \rightarrow T_2$. We accomplished this by raising the dissociation constant d_{33}, which destabilized the intermediate homodimer T_3T_3 relative to the heterodimer T_1T_3, and by lowering the dissociation constant $\langle a \rangle_{213}$, which stabilized the intermediate $E_2NT_1T_3$. An

alternate approach is to stabilize the homodimer T_1T_3 by lowering the appropriate dissociation constant d_{13}.

Computer simulation of the gate yielded the graph shown in Figure 1e. For the "1,1" case the output is very low, while in all three other cases it is high. We note that the output for the "1,0" case (introducing T_1) lags behind the other two ON cases. This is due to pre-equilibrium formation (see Experimental Section) of the T_1T_3 heterodimer at the expense of the T_3T_3 homodimer, causing a delay in T_3 autocatalysis. It is important to note that the NAND gate constructed here can also function as a one-input NOT gate. This is achieved when E_2 is permanently set to "1" and T_1 is the input. The significance of having a NOT function available for wiring together several gates will be discussed below.

The XOR gate: The XOR (Excluded OR) gate is a required participant in various logic devices and computational modules, including the half adder, half subtractor, full adder, and full subtractor, and has also been postulated to be central to the origin of life.[94] The XOR is not a trivial gate; its output is "1" when either one of the inputs is present, but is "0" when neither or both of the inputs are present (Table 1). The XOR gate implemented here is constructed similarly to the OR gate, but contains two additional competing heterodimeric cross-catalytic building blocks (Scheme 3f). Specifically, each of the homodimers T_1T_1 and T_2T_2 catalyzes the formation of T_3, while the heterodimer formed between T_1 and T_2 (T_1T_2) catalyzes E_1 and N, and E_2 and N, to form T_1 and T_2, respectively. Since the last two catalytic pathways (shown in Scheme 3f by thicker lines) are more efficient than the first two pathways, the production of T_1 and T_2 outraces the production of T_3 if both T_1 and T_2 are initially present at high concentrations. It should be noted though that it is not necessary to have both of the last two pathways competing with T_1 production; even one competing pathway from T_1T_2 is enough to produce the XOR function, as shown in the Computational Modules section.

The successful XOR design is achieved by weakening the forward catalytic pathways $T_1T_1 \rightarrow T_3$ and $T_2T_2 \rightarrow T_3$, while augmenting the backward competing catalytic pathways $T_1T_2 \rightarrow T_1$ and $T_1T_2 \rightarrow T_2$. This is done by destabilizing the intermediate dimers T_1T_1 and T_2T_2, namely by raising the dissociation constants d_{11} and d_{22}, and by simultaneously stabilizing the intermediate heterodimer T_1T_2 by lowering the dissociation rate constant d_{12}. It also helped to lower the dissociation constants $\langle a \rangle_{112}$ and $\langle a \rangle_{212}$, thus stabilizing the intermediates $E_1NT_1T_2$ and $E_2NT_1T_2$. Additionally, it was necessary to lower the background reaction rates (g_i), and since the ratio of the template-induced ligation rate to the background ligation rate was kept constant (see Experimental Section); this also lowers the rates of template-induced ligation (b_i) with no adverse effects. Computer simulation of the XOR gate (Figure 1f) reveals the expected "1" output for the "1,0" and "0,1" cases. The formation of T_3 is relatively slow in this case—resulting in maximal ON response of ~10 μM—due to the background production of T_1 and T_2. The output of the "1,1" case is "0", since T_1 and T_2 are formed quite early via the competing cross-catalytic pathways, inhibiting T_3 production.

The XNOR gate: The XNOR logic can be regarded as an Excluded NOR or as a combination of NOT and XOR. Its chemical implementation here follows the first option. Accordingly, we constructed a gate like the NOR, but with an additional heterodimeric cross-catalytic pathway that competes with the two pathways inhibiting T_3 autocatalysis in the NOR gate (Scheme 2g). As in the NOR gate, E_1 or E_2 were used as inputs. When either E_1 or E_2 are present the gate behaves like the NOR, while if both E_1 and E_2 are present the additional pathway provides an alternate mechanism for the production of T_3.

The key to getting this gate to work correctly is to keep the background reaction low, and to lower the relevant dissociation constants for the cross-catalytic reactions, thus strengthening these competing pathways (see Experimental Section). Computer simulation of the XNOR yielded the graph shown in Figure 1g. At about 15 minutes the system saturates as the limited quantities of reactants get used up, so it is important to observe the system's behavior before saturation sets in.

The NIFNOT gate: The last two-input logic gate we present in this section is the NIFNOT gate, which logically is just the Not of the IFNOT, as listed in Table 1. We designed the NIFNOT in Scheme 3h using an autocatalytic building block and three homodimeric cross-catalytic blocks. The cross-catalytic pathways are placed cyclically with increasing strength. Thus, T_3 catalyzes itself as well as T_1, T_1 catalyzes T_2, and T_2 catalyzes T_3. The idea behind this chain reaction is that if E_1 is not present, T_3 will be formed by autocatalysis. If E_1 is present but E_2 is not present, T_1 will be formed at the expense of T_3. However, if both E_1 and E_2 are present, T_1 will then catalyze T_2, which will in turn catalyze T_3. So only the combination of E_1 without E_2 results in the absence of T_3.

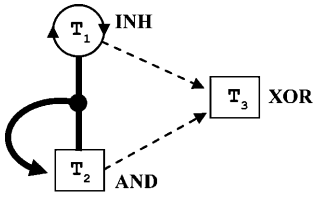
Successful implementation of this gate depends on low background reaction, stable intermediate dimers T_1T_1 and T_2T_2, and stable intermediates $E_2NT_1T_1$, $E_3NT_2T_2$ and $E_1NT_3T_3$ (see Experimental Section). Computer simulation of this NIFNOT yielded the graph shown in Figure 1h. We present the results for the first 15 minutes only, since this is the relevant part and afterwards the system goes into saturation.

Computational modules: Logic gates may be joined together into simple arithmetic units such as the full adder, half adder, full subtractor and half subtractor.[65,66] The integration of the gates can be achieved in chemical systems by two different methods. The first makes use of multifunctional molecules that exhibit different logic in response to different stimuli (e.g., different light wavelengths). In the second method one follows two or more molecules in parallel. While each of the molecules performs a single Boolean

logic, the overall product of the network is the expected arithmetic.

A half adder is a simple logical circuit that performs an addition operation on two binary digits. It has two inputs and two outputs, the sum S and the carry C. S and C are the two-bit XOR and AND operations, respectively, of the same two inputs. A half subtractor circuit subtracts one bit from another and places the difference D and the borrow B. It thus implements INHIBIT and XOR logics that share the same two inputs.

In our case, we note that the network topology that facilitates XOR gate formation of T_3 (Scheme 3 f) simultaneously produces the AND outputs in T_1 and T_2. Thus we have produced the half adder. Continuing this approach, we have constructed an "asymmetric" network (Scheme 4), in which the heterodimer T_1T_2 catalyzes only T_2 and not T_1, while T_1 can be produced autocatalytically. As a result, the T_3 output is a XOR, while the T_1 output is an INHIBIT and the T_2 output is an AND. Thus we are able to design simultaneously half subtractor (T_1 and T_3) and half adder (T_2 and T_3) arithmetic within a single network (Figure 2).



Scheme 4. Graphical description of an alternate XOR gate in T_3 that simultaneously produces an INHIBIT in T_1 and an AND in T_2, allowing the construction of a half subtractor and half adder. The dashed lines represent especially weak catalytic pathways.

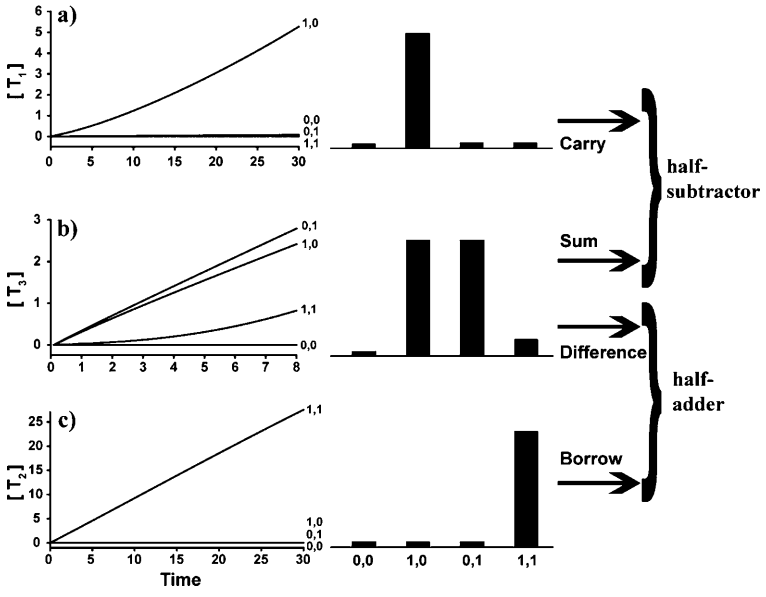
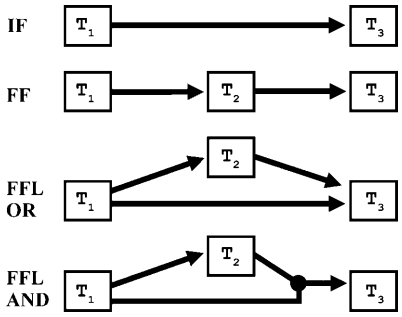


Figure 2. Kinetic simulations of the network shown in Scheme 4. Production of a) T_1, b) T_3, and c) T_2, in μM, as a function of time (min), yielding INHIBIT, XOR, and AND logic, respectively. Panels a) and b) analyzed together are equivalent to the half subtractor arithmetic unit, while b) and c) together are equivalent to the half adder. The initial concentrations of the starting materials E_1, E_2, E_3, and N were each 100 μM, T_3 was initialized at 0 μM, and the inputs were T_1 and T_2 , for which a "1" input corresponds to 10 μM.

We note that the desired network topology is more difficult to construct than the previous XOR, as its dynamics essentially represent a delicate balancing act among the various pathways. It requires assigning very low background rates and a low dissociation rate constant d_{12} in order to stabilize the intermediate dimer T_1T_2 and strengthen the catalytic pathway $T_1T_2 \rightarrow T_2$ (see Experimental Section). Computer simulation of the dynamics of this network yielded the graphs shown in Figure 2, for T_1, T_3, and T_2. The production of T_1 and T_3 is lower than that of T_2, because the pathways producing them are weaker than the one producing T_2. Additionally, in this topology, the XOR gate works only for short times of up to ~20 minutes; after that it transforms into the simple OR logic. It seems that producing this gate would be an interesting challenge for future experiments, as it may represent the maximal "complexity" inherent in such networks.

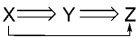
Network motifs: Specific modules found highly frequently within complex networks were given the general term "network motifs". This concept was used to detect basic building blocks of cellular networks, with a special focus on gene regulatory networks.[4,6,7,95] The studied network motifs, such as the feed-forward loop (FFL), include auto or cross regulation of small sub-networks that consist of three nodes. They thus represent a slightly higher level of complexity relative to the logic gates. No one has yet explored the network motifs on chemical systems, and to do so we extended our study to include motifs that are similar to the FFL motifs. Their design can be readily achieved using the same network building blocks, and simulated with the same equations and computations, as described above.

One of the most common examples in biological systems is the "coherent type-1" feed-forward loop network motif.[7] This motif appears in two forms, as it may be joined at the end by an OR gate or by an AND gate. It has been suggested that the FFLs have some unique features, such as acceleration and pulse generation, which cannot be carried out by cascades or simple regulation. The basic schematic diagram for the coherent FFL is shown in Equation (3). Implementation of the two coherent FFL forms on our catalytic network is shown in Scheme 5. In both cases, T_1 catalyzes the formation of T_2. Subsequently, in the FFL OR case, either T_1 or T_2 may each catalyze the formation of T_3, as in the OR gate. In the FFL AND case, both T_1 and T_2 are required to catalyze



Scheme 5. Graphical descriptions of the feed-forward loop and its variations: the direct IF gate, the simple cascade "feed-forward" (FF) logic, and the "coherent type-1" feed-forward loop network motif joined at the end by an OR gate (FFL OR) or by an AND gate (FFL AND).

the formation of T_3, as in the AND gate. We note that T_2 serves as both the output of T_1 and the input of T_3, which requires efficient template release.



(3)

We explored the behavior of these FFL network motifs in comparison with two control functions: the direct IF gate, in which T_1 directly catalyzes the formation of T_3, and a simple cascade "feed-forward" (FF) logic, in which T_1 indirectly catalyzes the formation of T_3 by first catalyzing the formation of T_2, which in turn catalyzes the formation of T_3. Both of these functions are also shown in Scheme 5. The FF is a concatenation of two YES gates, while the FFL OR motif is a concatenation of a YES with an OR, and the FFL AND motif is a concatenation of a YES with an AND.

To study the effect of the different motifs on the kinetic profile of T_3 production, we ran computer simulations of these functions under three different sets of parameters. Our computer program and GUI (see Experimental Section) allow us to use multiple inputs and thus to execute simultaneous runs of the network motifs and their controls (Figure 3).

In the first case (Figure 3a) the system was simulated using the default parameters (except with higher $\langle a \rangle$; see Experimental Section), that is, those with reaction constants most closely corresponding to realistic laboratory values. Here, the FFL OR and the IF initially yield similar results, while the FFL AND has a slightly delayed reaction, and the FF is delayed even more. Eventually, the FFL AND and FF catch up with the FFL OR, while the direct IF falls behind. Essentially, all the feed-forward motifs are alternatives that "amplify" the IF. Of these, the FFL OR is the strongest, since it utilizes an additional pathway relative to the IF or FF.

In the second case, the simulation parameters were modified to magnify the formation of T_3 through the FFL OR motif (Figure 3b). To achieve this, efficient $T_2T_2 \rightarrow T_3$ and weak $T_1T_2 \rightarrow T_3$ catalytic pathways were utilized, by assigning

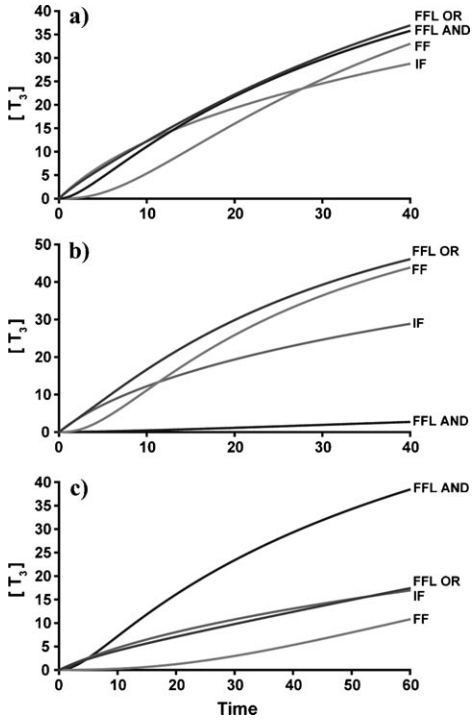


Figure 3. Kinetic simulations of the FFL network motifs in comparison with the simpler IF and FF motifs. Production of T_3 (μM) as a function of time (min) is shown for "ON" T_1 input (10 μM) and initial zero concentration of T_2. Three sets of parameters were used to simulate the behavior of the network a) under default conditions, b) under conditions that facilitate substantial enhancement of T_3 production through the FFL OR functionality, or c) under conditions that facilitate substantial enhancement of T_3 production through the FFL AND functionality. The initial concentrations of E_1, E_2, E_3, and N were each set at 100 μM, and T_3 was initialized at zero.

low d_{22} and very high d_{12}, respectively. The kinetic behavior of the FFL OR is now maximized both at the beginning of the reaction, at which its initial rate is as high as the IF function, and during the later stages, at which it is faster than the FF motif. The FFL AND motif is totally inactive in these conditions, as expected when assigning high d_{12}.

In the third case the tables have turned, and the FFL AND is more effective than the other motifs (Figure 3c). The simulation of this scenario is achieved by reducing the efficiency of all catalytic pathways except for $T_1T_2 \rightarrow T_3$, by assigning d_{11} and d_{22} higher than the defaults. While the FFL AND is clearly improved, the FFL OR and the IF operate similarly to the default case, and the FF is practically ineffective.

From a design point of view, the FFL OR motif is a practical way to "amplify" an IF gate. While both the FF and FFL AND motifs serve to introduce an initial delay in an otherwise straightforward logic, the FFL AND is much faster to recover, providing a mechanism for "pulse generation".

Conclusion

We have shown in this paper how basic auto- and cross-catalytic processes may be connected and combined to form logic gates, simple arithmetic units and network motifs, yielding various forms of "systems chemistry" through competitive or cooperative organized behavior.

The ability to integrate molecular logic gates into even more complex functionality and modular multilevel circuits is crucial for the realization of molecular computation. One approach is to link together different logic gates by concatenation,[96,97] that is, by using the outputs of one gate as the inputs of a second gate, thus allowing the cascading of information down a stream as in semiconductor logic. This requires qualitative and quantitative input–output homogeneity,[65,66] which is not always present in molecular logic. We note that several of the logic gates presented here do fulfill this requirement. Specifically, the OR, AND, NOT (as part of the NAND logic) and XOR gates, the inputs and outputs of which are templates and not reactants, allow the wiring of one gate with another. These gates alone, if properly linked together and re-used, are sufficient to produce all possible logic operations, and also more complicated arithmetic, as required for example in the full adder.

We claim that our computer simulations provide actual "recipes" for chemically synthesizing these complex functional motifs. These guidelines can now be applied to various systems known to contain template-assisted processes, such as peptides, nucleotides, and organic compounds.

Experimental Section

Computational methods: Our computations were performed by simulating, in Matlab, the kinetics of the basic auto and cross-catalytic chemical reactions, as defined in Equations (1) and (2), for $i, j, k = 1, 2, 3$. For this purpose we built a user friendly GUI (graphical user interface), which allowed us to interactively adjust the initial concentrations of reactants and templates as well as the rate constants. This GUI also allowed us to enter multiple inputs for any (one or more) of the parameters, resulting in several runs and comparison of the results in one plot.

When initially called, the GUI displayed the following default parameter values:

a, $\langle d \rangle$,and $\langle f \rangle$, the diffusion constants: 10^9, 10^6 and 10^6, respectively.

b_i, the reaction constants of the template-induced ligation: 10.

g_i, the reaction constants of the background ligation: 1.

d_{ijk}, the dissociation constants of the intermediate dimers T_jT_k: 10.

$\langle a \rangle_{ijk}$, the dissociation constants of the intermediates $E_iNT_jT_k$: 10.

f_{ijk}, the dissociation constants of the intermediate trimers $T_iT_jT_k$: 10^3.

These defaults correspond to realistic laboratory values. The values of g_i and $\langle a \rangle_{ijk}$ were not explicitly displayed by the GUI. Instead, g_i was set to be proportional to b_i, and $\langle a \rangle_{ijk}$ was set to be proportional to the square of f_{ijk}, as both these pairs relate chemically similar parameters. It is the constants of proportionality that were displayed and could be adjusted within the GUI. For this reason, when changing g_i and $\langle a \rangle_{ijk}$, we usually kept the proportionality and simultaneously adjusted b_i or f_{ijk}.

Each specific gate, module, or motif was run by setting up the appropriate network, that is, by allowing or interrupting the appropriate network connections. For example, a productive autocatalytic pathway producing T_i corresponded to a nominal value for the dissociation constant $\langle a \rangle_{iii}$, a productive homodimeric cross-catalytic pathway from T_j to T_i corresponded to a nominal value for the dissociation constant $\langle a \rangle_{ijj}$, and a productive heterodimeric cross-catalytic pathway from T_j and T_k to T_i corresponded to a nominal value for the dissociation constant $\langle a \rangle_{ijk}$. The interruption of any of these connections corresponded to a very high value for the appropriate $\langle a \rangle_{ijk}$. In practice, we ran our simulations using the following approximation: for numerical reasons, we kept $\langle a \rangle_{ijk}$ to be proportional to the square of f_{ijk}, and zeroed the specific a_{ijk} values corresponding to the interrupted network conditions.

Table 2 lists the initial components and inputs for each case, as well as the parameter values that differed from the defaults, in order to strengthen or weaken certain catalytic pathways, or as "experimental conditions" in order to simulate actual laboratory conditions:

When the "Run" button was activated, the kinetics of Equations (1) and (2) were simulated. First the initial concentrations of the intermediates was calculated, by running the reactions of Equation (1) without ligation until equilibrium was reached. Then all stages of Equations (1) and (2) were simultaneously computed. By using constant incremental time steps, we calculated at each time step the extent of each reaction, that is, the product of the rate constants and the reactants. After all these were calculated, the concentrations of all reactants and products were appropriately adjusted. This procedure was repeated continually until the maximum given time was reached. Mathematically, this is equivalent to solving a set of differential equations using the Euler method.

Acknowledgement

We thank Zehavit Dadon for her assistance in writing the manuscript. G.A. thanks the Human Frontier Science Program for Career Development Award. This research is supported by a grant from the Israel Science Foundation.

[1] L. H. Hartwell, J. J. Hopfield, S. Leibler, A. W. Murray, *Nature* **1999**, *402*, C47.
[2] H. Jeong, B. Tombor, R. Albert, Z. N. Oltval, A. L. Barabasi, *Nature* **2000**, *407*, 651.
[3] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, A. L. Barabasi, *Science* **2002**, *297*, 1551.
[4] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, U. Alon, *Science* **2002**, *298*, 824.
[5] J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. M. Walhout, M. E. Cusick, F. P. Roth, M. Vidal, *Nature* **2004**, *430*, 88.
[6] U. Alon, *Nat. Rev. Genet.* **2007**, *8*, 450.
[7] U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits*, Chapman and Hall, **2007**.
[8] G. R. L. Cousins, S.-A. Poulsen, J. K. M. Sanders, *Curr. Opin. Chem. Biol.* **2000**, *4*, 270.
[9] J.-M. Lehn, A. V. Eliseev, *Science* **2001**, *291*, 2331.
[10] O. Ramstrom, J.-M. Lehn, *Nat. Rev. Drug Discovery* **2002**, *1*, 26.
[11] J.-L. Reymond, *Angew. Chem.* **2004**, *116*, 5695; *Angew. Chem. Int. Ed.* **2004**, *43*, 5577.
[12] Z. J. Gartner, *Pure Appl. Chem.* **2006**, *78*, 1.
[13] B. de Bruin, P. Hauwert, J. N. H. Reek, *Angew. Chem.* **2006**, *118*, 2726; *Angew. Chem. Int. Ed.* **2006**, *45*, 2660.
[14] P. T. Corbett, J. Leclaire, L. Vial, K. R. West, J.-L. Wietor, J. K. M. Sanders, S. Otto, *Chem. Rev.* **2006**, *106*, 3652.
[15] K. Severin, *Chem. Eur. J.* **2004**, *10*, 2565.
[16] P. T. Corbett, J. K. M. Sanders, S. Otto, *Angew. Chem.* **2007**, *119*, 9014; *Angew. Chem. Int. Ed.* **2007**, *46*, 8858.
[17] G. Von Kiedrowski, *Angew. Chem.* **1986**, *98*, 932; *Angew. Chem. Int. Ed. Engl.* **1986**, *25*, 932.
[18] W. S. Zielinski, L. E. Orgel, *Nature* **1987**, *327*, 346.
[19] L. E. Orgel, *Nature* **1992**, *358*, 203.
[20] T. Li, K. C. Nicolaou, *Nature* **1994**, *369*, 218.

[21] N. Paul, G. F. Joyce, *Proc. Natl. Acad. Sci. USA* **2002**, *99*, 12733.
[22] D. H. Lee, J. R. Granja, J. A. Martinez, K. Severin, M. R. Ghadiri, *Nature* **1996**, *382*, 525.
[23] K. Severin, D. H. Lee, J. A. Martinez, M. R. Ghadiri, *Chem. Eur. J.* **1997**, *3*, 1017.
[24] S. Yao, I. Ghosh, R. Zutshi, J. Chmielewski, *J. Am. Chem. Soc.* **1997**, *119*, 10559.
[25] S. Yao, I. Ghosh, R. Zutshi, J. Chmielewski, *Angew. Chem.* **1998**, *110*, 489; *Angew. Chem. Int. Ed.* **1998**, *37*, 478.
[26] M. Levy, A. D. Ellington, *J. Mol. Evol.* **2003**, *56*, 607.
[27] E. A. Wintner, J. J. Rebek, *Acta Chim. Scand.* **1996**, *50*, 469.
[28] B. Wang, I. O. Sutherland, *Chem. Commun.* **1997**, 1495.
[29] M. Kindermann, I. Stahl, M. Reimold, W. M. Pankau, G. von Kiedrowski, *Angew. Chem.* **2005**, *117*, 6908; *Angew. Chem. Int. Ed.* **2005**, *44*, 6750.
[30] E. Kassianidis, D. Philp, *Angew. Chem.* **2006**, *118*, 6492; *Angew. Chem. Int. Ed.* **2006**, *45*, 6344.
[31] D.-E. Kim, G. F. Joyce, *Chem. Biol.* **2004**, *11*, 1505.
[32] M. Levy, A. D. Ellington, *Proc. Natl. Acad. Sci. USA* **2003**, *100*, 6416.
[33] E. Kassianidis, D. Philp, *Chem. Commun.* **2006**, 4072.
[34] K. Severin, D. H. Lee, J. A. Martinez, M. Vieth, M. R. Ghadiri, *Angew. Chem.* **1998**, *110*, 133; *Angew. Chem. Int. Ed.* **1998**, *37*, 126.
[35] G. Ashkenasy, M. R. Ghadiri, *J. Am. Chem. Soc.* **2004**, *126*, 11140.
[36] D. Sievers, G. von Kiedrowski, *Nature* **1994**, *369*, 221.
[37] D. Sievers, G. von Kiedrowski, *Chem. Eur. J.* **1998**, *4*, 629.
[38] S. Yao, I. Ghosh, R. Zutshi, J. Chmielewski, *Nature* **1998**, *396*, 447.
[39] A. Saghatelian, Y. Yokobayashi, K. Soltani, M. R. Ghadiri, *Nature* **2001**, *409*, 797.
[40] T. Achilles, G. Von Kiedrowski, *Angew. Chem.* **1993**, *105*, 1225; *Angew. Chem. Int. Ed.* **1993**, *32*, 1198.
[41] G. Ashkenasy, R. Jagasia, M. Yadav, M. R. Ghadiri, *Proc. Natl. Acad. Sci. USA* **2004**, *101*, 11872.
[42] D. H. Lee, K. Severin, M. R. Ghadiri, *Curr. Opin. Chem. Biol.* **1997**, *1*, 491.
[43] A. Robertson, A. J. Sinclair, D. Philp, *Chem. Soc. Rev.* **2000**, *29*, 141.
[44] R. Issac, Y. W. Ham, J. Chmielewski, *Curr. Opin. Struct. Biol.* **2001**, *11*, 458.
[45] I. Ghosh, J. Chmielewski, *Curr. Opin. Chem. Biol.* **2004**, *8*, 640.
[46] Z. Dadon, N. Wagner, G. Ashkenasy, *Angew. Chem.* **2008**, *120*, 6221; *Angew. Chem. Intl. Ed.* **2008**, *47*, 6128.
[47] G. von Kiedrowski, *Bioorg. Chem. Front.* **1993**, *3*, 113.
[48] S. Assouline, S. Nir, N. Lahav, *J. Theor. Biol.* **2001**, *208*, 117.
[49] D. N. Reinhoudt, D. M. Rudkevich, F. de Jong, *J. Am. Chem. Soc.* **1996**, *118*, 6880.
[50] T. Tjivikua, P. Ballester, J. Rebek, Jr., *J. Am. Chem. Soc.* **1990**, *112*, 1249.
[51] F. M. Menger, A. V. Eliseev, N. A. Khanjin, M. J. Sherrod, *J. Org. Chem.* **1995**, *60*, 2870.
[52] P. R. Wills, S. A. Kauffman, B. M. R. Stadler, P. F. Stadler, *Bull. Math. Biol.* **1998**, *60*, 1073.
[53] B. M. R. Stadler, P. F. Stadler, P. Schuster, *Bull. Math. Biol.* **2000**, *62*, 1061.
[54] J. Rivera Islas, V. Pimienta, J.-C. Micheau, T. Buhse, *Biophys. Chem.* **2003**, *103*, 201.
[55] J. Rivera Islas, V. Pimienta, J.-C. Micheau, T. Buhse, *Biophys. Chem.* **2003**, *103*, 191.
[56] J. Rivera Islas, J.-C. Micheau, T. Buhse, *Origins Life Evol. Biosphere* **2004**, *34*, 497.
[57] E. Peacock-Lopez, D. B. Radov, C. S. Flesner, *Biophys. Chem.* **1997**, *65*, 171.
[58] K. M. Beutel, E. Peacock-Lopez, *J. Chem. Phys.* **2006**, *125*, 024908/1.
[59] K. M. Beutel, E. Peacock-Lopez, *J. Chem. Phys.* **2007**, *126*, 125104/1.
[60] R. F. Ludlow, S. Otto, *Chem. Soc. Rev.* **2008**, *37*, 101.
[61] G. Ashkenasy, M. R. Ghadiri, in *Understanding Biology Using Peptides: Proceedings of the 19th American Peptide Symposium*, **2006**, pp. 645.
[62] A. Luther, R. Brandsch, G. von Kiedrowski, *Nature* **1998**, *396*, 245.
[63] R. Issac, J. Chmielewski, *J. Am. Chem. Soc.* **2002**, *124*, 6808.
[64] X. Li, J. Chmielewski, *J. Am. Chem. Soc.* **2003**, *125*, 11820.
[65] A. P. de Silva, *Nat. Mater.* **2005**, *4*, 15.
[66] A. P. de Silva, S. Uchiyama, *Nat. Nanotech.* **2007**, *2*, 399.
[67] A. Credi, *Angew. Chem. 2007, 119, 5568; Angew. Chem. Int. Ed.* **2007**, *46*, 5472; *Angew. Chem. Int. Ed.* **2007**, *46*, 5472.
[68] D. Gust, T. A. Moore, A. L. Moore, *Chem. Commun.* **2006**, 1169.
[69] U. Pischel, *Angew. Chem.* **2007**, *119*, 4100; *Angew. Chem. Int. Ed.* **2007**, *46*, 4026.
[70] A. P. de Silva, T. P. Vance, M. E. S. West, G. D. Wright, *Org. Biomol. Chem.* **2008**, *6*, 2468.
[71] K. Szacilowski, *Chem. Rev.* **2008**, *108*, 3481.
[72] D. Margulies, C. E. Felder, G. Melman, A. Shanzer, *J. Am. Chem. Soc.* **2007**, *129*, 347.
[73] D. Margulies, G. Melman, A. Shanzer, *Nat. Mater.* **2005**, *4*, 768.
[74] D. Margulies, G. Melman, A. Shanzer, *J. Am. Chem. Soc.* **2006**, *128*, 4865.
[75] C. C. Guet, M. B. Elowitz, W. Hsing, S. Leibler, *Science* **2002**, *296*, 1466.
[76] J. E. Dueber, B. J. Yeh, R. P. Bhattacharyya, W. A. Lim, *Curr. Opin. Struct. Biol.* **2004**, *14*, 690.
[77] J. E. Dueber, B. J. Yeh, K. Chak, W. A. Lim, *Science* **2003**, *301*, 1904.
[78] F. J. Isaacs, D. J. Dwyer, J. J. Collins, *Nat. Biotech.* **2006**, *24*, 545.
[79] R. McDaniel, R. Weiss, *Curr. Opin. Biotechnol.* **2005**, *16*, 476.
[80] S. A. Benner, A. M. Sismour, *Nat. Rev. Genet.* **2005**, *6*, 533.
[81] H. Lederman, J. Macdonald, D. Stefanovic, M. N. Stojanovic, *Biochemistry* **2006**, *45*, 1194.
[82] M. N. Stojanovic, S. Semova, D. Kolpashchikov, J. Macdonald, C. Morgan, D. Stefanovic, *J. Am. Chem. Soc.* **2005**, *127*, 6914.
[83] M. N. Stojanovic, D. Stefanovic, *Nat. Biotech.* **2003**, *21*, 1069.
[84] M. N. Stojanovic, D. Stefanovic, *J. Am. Chem. Soc.* **2003**, *125*, 6673.
[85] M. N. Stojanovic, T. E. Mitchell, D. Stefanovic, *J. Am. Chem. Soc.* **2002**, *124*, 3555.
[86] G. Seelig, D. Soloveichik, D. Y. Zhang, E. Winfree, *Science* **2006**, *314*, 1585.
[87] N. C. Gianneschi, M. R. Ghadiri, *Angew. Chem.* **2007**, *119*, 4029; *Angew. Chem. Int. Ed.* **2007**, *46*, 3955.
[88] A. Saghatelian, N. H. Voelcker, K. M. Guckian, V. S. Y. Lin, M. R. Ghadiri, *J. Am. Chem. Soc.* **2003**, *125*, 346.
[89] H. Yan, L. Feng, T. H. LaBean, J. H. Reif, *J. Am. Chem. Soc.* **2003**, *125*, 14246.
[90] C. Mao, T. H. LaBean, J. H. Reif, N. C. Seeman, *Nature* **2000**, *407*, 493.
[91] A. P. de Silva, N. D. McClenaghan, *Chem. Eur. J.* **2004**, *10*, 574.
[92] H. T. Baytekin, E. U. Akkaya, *Org. Lett.* **2000**, *2*, 1725.
[93] F. M. Raymo, S. Giordani, *Proc. Natl. Acad. Sci. USA* **2002**, *99*, 4941.
[94] G. W. Rowe, *Theoretical Models in Biology: The Origin of Life, the Immune System, and the Brain*, Oxford University Press, Oxford, **1994**.
[95] S. Shen-Orr, R. Milo, S. Mangan, U. Alon, *Nat. Genet.* **2002**, *31*, 64.
[96] T. Niazov, R. Baron, E. Katz, O. Lioubashevski, I. Willner, *Proc. Natl. Acad. Sci. USA* **2006**, *103*, 17160.
[97] N. H. Voelcker, K. M. Guckian, A. Saghatelian, M. R. Ghadiri, *Small* **2008**, *4*, 427.

Received: September 4, 2008
Published online: December 23, 2008

