Chemical mass-action systems as analog computers: implementing arithmetic computations at specified speed

Badal Joshi
California State University San Marcos

Formal Reaction Kinetics

May 7, 2024

Computation with reaction networks

- Computation near action.
- Biological cells:
  - process information, and
  - have macromolecules.
- How to compute in a wet environment of a living cell?
- Synthetic biologists can implement reaction networks, using for instance DNA-strand displacement.

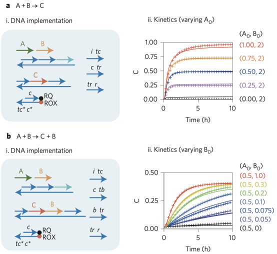
nature nanotechnology

ARTICLES

PUBLISHED ONLINE: 29 SEPTEMBER 2013 | DOI: 10.1038/NNANO.2013.189

Programmable chemical controllers made from DNA

Yuan-Jyue Chen[1], Neil Dalchau[2], Niranjan Srinivas[3], Andrew Phillips[2], Luca Cardelli[2], David Soloveichik[4]* and Georg Seelig[1,5]*



DNA as a universal substrate for chemical kinetics

David Soloveichik[a,1], Georg Seelig[a,b,1], and Erik Winfree[c,1]

A Ideal chemical reactions

$$1:\ X_1 + X_2 \xrightarrow{k_1} 2X_2$$

$$2:\ X_1 \xrightarrow{k_2} 2X_1$$

$$3:\ X_2 \xrightarrow{k_3} \emptyset$$

	unscaled	scaled
k_1	1.5	$5\cdot10^5$ /M/s
k_2	1	1/300 /s
k_3	1	1/300 /s

B DNA reaction modules

1:
$$X_1 + L_1 \underset{q_{max}}{\overset{q_1}{\rightleftharpoons}} H_1 + B_1$$
$$X_2 + H_1 \xrightarrow{q_{max}} O_1$$
$$O_1 + T_1 \xrightarrow{q_{max}} 2X_2$$

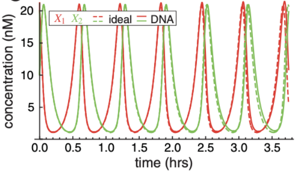
buffering module
$$X_2 + LS_2 \underset{q_{max}}{\overset{qs_2}{\rightleftharpoons}} HS_2 + BS_2$$

2:
$$X_1 + G_2 \xrightarrow{q_2} O_2$$
$$O_2 + T_2 \xrightarrow{q_{max}} 2X_1$$

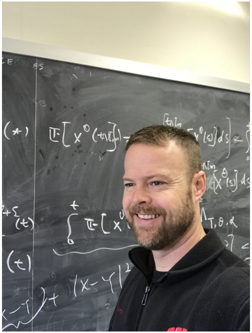
3:
$$X_2 + G_3 \xrightarrow{q_3} O_3$$

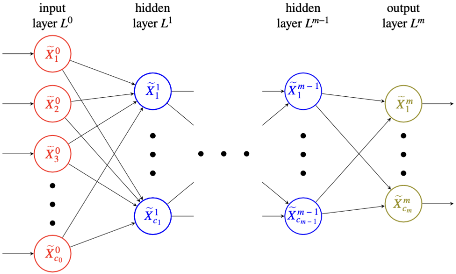
$$q_1 = \gamma^{-1} k_1$$
$$q_2 = \gamma^{-1} k_2 C_{max}^{-1}$$
$$q_3 = \gamma^{-1} k_3 C_{max}^{-1}$$
$$qs_2 = \gamma^{-1}(\sigma - \sigma_2)$$

C Simulation of ideal and DNA reactions



Joint work with David Anderson (University of Wisconsin-Madison)





- David F. Anderson, Badal Joshi, and Abhishek Deshpande. On reaction network implementations of neural networks, *Journal of Royal Society Interface*, 18, 177, (2021).
- David F. Anderson, and Badal Joshi. Chemical mass-action systems as analog computers: implementing arithmetic computations at specified speed, arXiv, April 2024.

Why analog computing?

Initial value problem

$$\dot{x}(t) = 1 - x(t), \quad x(0) = x_0.$$

Reaction network "computer"

$$0 \rightleftarrows X \qquad x(0) = x_0$$

What about arithmetic or discrete processes?

Existing schemata for arithmetic: Conservation law based schema

- *Conservation-law based schema* [1]: eg. Addition
  - $a(0) = a, \quad b(0) = b, \quad x(0) = 0.$

$$A \to X, \quad B \to X.$$
$$\lim_{t\to\infty} x(t) = a + b.$$

- Works for discrete (marbles) or continuous (water); stochastic or deterministic.
- *Memory lost.* No trace of inputs remains after computation. Problem when composing elementary computations or reusing inputs.

[1] H. L. Chen, D. Doty, W. Reeves, D. Soloveichik. "Rate-independent computation in continuous chemical reaction networks." Journal of the ACM 70.3 (2023): 1-61.

Existing schemata for arithmetic: Positive steady state based schema

- *Positive steady state based schema*[2]: eg. Addition

$$A \to A + X, \quad B \to B + X, \quad X \to 0,$$
$$\dot{x}(t) = a(t) + b(t) - x(t),$$
$$\lim_{t\to\infty} x(t) = \lim_{t\to\infty} (a(t) + b(t)) \,.$$

- If this is the entire network:

$$a(t) = a(0), \quad b(t) = b(0) \text{ for all } t \geq 0,$$
$$\lim_{t\to\infty} x(t) = a(0) + b(0).$$

- Inputs not degraded.
- Can do *parallel computation*.

[2] H. J. Buisman, H.M. ten Eikelder, P.A. Hilbers, & A.M. Liekens (2009). Computing algebraic functions with biochemical reaction networks. Artificial life, 15(1), 5-19.

Reaction network and mass action kinetics

$$
\begin{aligned}
X_1 + X_2 &\to X_3, \\
X_3 &\to X_1 + X_2, \\
2X_3 &\to 0.
\end{aligned}
$$

$$
\dot{x}(t) = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = c_1 x_1 x_2 \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} + c_2 x_3 \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} + c_3 x_3^2 \begin{pmatrix} 0 \\ 0 \\ -2 \end{pmatrix}.
$$

- A polynomial dynamical system
  $\dot{x} = f(x), \quad x \in \mathbb{R}^n, \quad f : \mathbb{R}^n \to \mathbb{R}^n$
  is a mass action system if and only if x_i appears in any monomial of $f_i(x)$ that has a negative coefficient.
- This condition ensures that the nonnegative orthant $\mathbb{R}^n_{\geq 0}$ is forward-invariant.

Negativity and dual rail representation

Dual rail map

$$a \mapsto (a_p, a_n) = \begin{cases} (a, 0), & \text{if } a > 0, \\ (0, -a), & \text{if } a \leq 0. \end{cases}$$

Most algebra in dual rail is straightforward.

$$\begin{aligned} a \cdot b &= (a_p, a_n) \cdot (b_p, b_n) = (a_p - a_n) \cdot (b_p - b_n) \\ &= a_p b_p + a_n b_n - a_p b_n - a_n b_p = (a_p b_p + a_n b_n, a_p b_n + a_n b_p). \end{aligned}$$

One exception: "real inversion"

$$a^{-1} = \begin{cases} \left(\frac{1}{a_p}, 0\right) & \text{if } a_p > 0, \\ \left(0, \frac{1}{a_n}\right) & \text{if } a_n > 0. \end{cases}$$

Assumptions

- able to record concentrations – inputs and outputs of computation – with arbitrary precision,
- can implement arbitrary reaction networks (no binary constraint),
- all reactions occur at the same rate.

Input independent speed of computation

Suppose we want reciprocal of $a > 0$.

$$0 \to X, \quad A + X \to A$$
$$\dot{x} = 1 - ax.$$

$$x(t) = \frac{1}{a} + \left(x(0) - \frac{1}{a}\right) e^{-at}.$$

- $x(t) \xrightarrow{t \to \infty} 1/a.$
- *Speed of computing n digits*:

$$\frac{n}{T_n} \sim a.$$

Input independent speed of computation

$$X \to 2X, \quad A + 2X \to A + X$$
$$\dot{x} = x(1 - ax).$$

Unique solution for all time:

$$x(t) = \frac{x_0/a}{(1/a - x_0)e^{-t} + x_0},$$

- $x(t) \xrightarrow{t \to \infty} 1/a.$
- *Speed of computing n digits*:

$$\frac{n}{T_n} \sim 1.$$

Speed of computation

Definition

Let $g : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a real number $g^* \in \mathbb{R}$. The *rate of convergence* of g to g^* is defined to be

$$\rho_g = -\limsup_{t\to\infty} \frac{\ln |g(t) - g^*|}{t},$$

whenever $\rho_g \in (0, \infty]$. Alternatively, in the context of computation, we say that $g(t)$ computes g^* at *speed* ρ_g.

$$|g(t) - g^*| \sim f(t)e^{-\rho_g t}, \qquad f(t) \text{ is sub-exponential.}$$

Simple convergence results

Lemma: For each of $i \in \{1, 2\}$, let $g_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to $g_i^* \in \mathbb{R}$ at a rate greater than ρ_{g_i}.

- The sum $g_1(t) + g_2(t)$ converges to $g_1^* + g_2^*$ at a rate that is at least

$$\min\{\rho_{g_1}, \rho_{g_2}\}.$$

- The product $g_1(t)g_2(t)$ converges to $g_1^* g_2^*$ at a rate that is at least

$$\begin{cases} \min\{\rho_{g_1}, \rho_{g_2}\}, & \text{if } g_1^* \neq 0, g_2^* \neq 0, \\ \rho_{g_1}, & \text{if } g_1^* = 0, g_2^* \neq 0, \\ \rho_{g_2}, & \text{if } g_1^* \neq 0, g_2^* = 0, \\ \rho_{g_1} + \rho_{g_2}, & \text{if } g_1^* = 0, g_2^* = 0. \end{cases}$$

Simple convergence results

Lemma: Let $g : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a nonzero real constant $g^* \in \mathbb{R} \setminus \{0\}$ at a rate that is at least ρ_g. Then $1/g(t)$ converges to $1/g^*$ at rate that is at least ρ_g.

Lemma: Let $g : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a non-negative constant $g^* \in \mathbb{R}_{\geq 0}$ at a rate that is at least ρ_g. Then for any $m \in \mathbb{R}_{>0}$, $g(t)^{1/m}$ converges to $(g^*)^{1/m}$ at rate that is at least

$$\begin{cases} \rho_g, & \text{if } g^* > 0, \\ \rho_g/m, & \text{if } g^* = 0. \end{cases}$$

Analysis of non-autonomous systems

<u>Lemma</u>: Let $g_1 : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to $g_1^* \in \mathbb{R}$ at a rate that is at least ρ_{g_1}; let $g_2 : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a positive limit $g_2^* \in \mathbb{R}_{>0}$ at a rate that is at least ρ_{g_2}. We assume that the g_1, g_2 are smooth enough so that for any $x(0) = x_0 \geq 0$, the following non-autonomous differential equation has a unique solution $x : \mathbb{R}_{\geq 0} \to \mathbb{R}$ for all time

$$\dot{x}(t) = g_1(t) - g_2(t)x(t).$$

Then $x(t)$ converges to g_1^*/g_2^* at rate that is at least

$$\min\{\rho_{g_1}, \rho_{g_2}, g_2^*\}.$$

Lemma: For $i \in \{1, 2\}$, let $g_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a positive limit $g_i^* \in \mathbb{R}_{>0}$ at a rate that is at least ρ_{g_i}. We assume that the g_1, g_2 are smooth enough so that for any $x(0) = x_0 > 0$, the following non-autonomous differential equation has a unique solution $x : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ for all time

$$\dot{x}(t) = x(t)(g_1(t) - g_2(t)x(t)^m), \qquad (m \in \mathbb{Z}_{>0}).$$

Then $x(t)$ converges to $(g_1^*/g_2^*)^{1/m}$ at rate that is at least

$$\min\{\rho_{g_1}, \rho_{g_2}, m g_1^*\}.$$

Lemma: Let $g_1 : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a negative limit $g_1^* \in \mathbb{R}_{<0}$ at a rate that is at least ρ_{g_1}; let $g_2 : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to a positive limit $g_2^* \in \mathbb{R}_{>0}$ at a rate that is at least ρ_{g_2}. We assume that g_1 and g_2 are smooth enough so that for any $x(0) = x_0 > 0$, the following non-autonomous differential equation has a unique solution $x : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ for all time

$$\dot{x}(t) = x(t)(g_1(t) - g_2(t)x(t)^m), \qquad (m \in \mathbb{Z}_{>0}).$$

Then $x(t)$ converges to 0 at rate that is at least

$$\min\{\rho_{g_1}, -g_1^*\}.$$

Analysis of non-autonomous systems

<u>Lemma</u>: Let $g : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a real-valued function that converges to 0 at a rate that is at least ρ_g. We assume that g is smooth enough so that for any $x(0) = x_0 > 0$, the following non-autonomous differential equation has a unique solution $x : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ for all time

$$\dot{x}(t) = x(t)(1 - g(t)x(t)^m), \qquad (m \in \mathbb{Z}_{>0}).$$

Then for any $\varepsilon > 0$, there is a $c_\varepsilon > 0$ such that

$$x(t) \geq c_\varepsilon e^{(\min\{\rho_g/m, 1\} - \varepsilon)t}, \text{ for all } t \geq 0.$$

Absolute difference Consider the reaction network and mass-action system

$$\begin{aligned}
&\{Y \to 2Y, \quad 2A+3Y \to 2A+2Y,\\
&2B+3Y \to 2B+2Y, \quad A+B+3Y \to A+B+5Y,\\
&X \to 2X, \quad Y+2X \to Y+X\},\\
&\dot{y}(t) = y(t)\left(1-(a(t)-b(t))^2 y(t)^2\right),\\
&\dot{x}(t) = x(t)\left(1-y(t)x(t)\right),
\end{aligned}$$

where $a(t)$ and $b(t)$ are non-negative functions of time that converge to non-negative constants a^* and b^* at rates that are at least ρ_a and ρ_b, respectively. Then the concentration of species X, i.e. the variable x, computes absolute difference $(a^*, b^*) \mapsto x^* = |a^* - b^*|$ at speed that is at least

$$\min\{\rho_a, \rho_b, 1\}.$$

Lemma (Rectified subtraction): Consider the reaction network and mass-action system

$$\{Y \to 2Y, \quad 2A + 3Y \to 2A + 2Y, \quad 2B + 3Y \to 2B + 2Y,$$
$$A + B + 3Y \to A + B + 5Y, \quad A + Y + X \to A + Y + 2X,$$
$$B + Y + X \to B + Y, \quad Y + 2X \to Y + X\},$$
$$\dot{y}(t) = y(t)\left(1 - (a(t) - b(t))^2 y(t)^2\right),$$
$$\dot{x}(t) = y(t)x(t)\left(a(t) - b(t) - x(t)\right),$$

where $a(t)$ and $b(t)$ are non-negative functions of time that converge to non-negative constants a^* and b^* at rates that are at least ρ_a and ρ_b, respectively. Then the concentration of species X, i.e. the variable x, computes rectified subtraction

$$(a^*, b^*) \mapsto \begin{cases} a^* - b^* & \text{if } a^* > b^* \\ 0 & \text{if } a^* \leq b^* \end{cases}$$

at speed that is at least

$$\min\{\rho_a, \rho_b, 1\}.$$

Lemma (Partial real inversion): Consider the reaction network and mass-action system

$$
\begin{aligned}
&\{Y \to 2Y, \quad A_p + 2Y \to A_p + Y, \\
&A_n + 2Y \to A_n + Y, \quad A_p + Y + X \to A_p + Y + 2X, \\
&A_n + Y + X \to A_n + Y, \quad 2A_p + Y + 2X \to 2A_p + Y + X\}, \\
&\dot{y}(t) = y(t)\left(1 - (a_p(t) + a_n(t))y(t)\right) \\
&\dot{x}(t) = y(t)x(t)\left(a_p(t)(1 - a_p(t)x(t)) - a_n(t)\right)
\end{aligned}
$$

We assume that one and only one of a_p^* or a_n^* is positive, while the other is zero. Then the concentration of species X, i.e. the variable x, computes partial real inversion

$$
(a_p^*, a_n^*) \mapsto \begin{cases} 1/a_p^*, & \text{if } a_p^* > 0, \\ 0, & \text{if } a_p^* = 0, \end{cases}
$$

at speed that is at least

$$
\min\{\rho_{a_p}, \rho_{a_n}, 1\}.
$$

m-th root Consider the reaction network and mass-action system

$$\begin{aligned}&\{Y \to 2Y, \quad A + (m+1)Y \to A + mY,\\ &X \to 2X, \quad Y + 2X \to Y + X\}\\ &\dot{y}(t) = y(t)\left(1 - a(t)y(t)^m\right),\\ &\dot{x}(t) = x(t)\left(1 - y(t)x(t)\right),\end{aligned}$$

where $m \in \mathbb{Z}_{\geq 2}$ and $a(t)$ is a non-negative-valued function of time that converges to a non-negative constant a^* at a rate that is at least ρ_a. Then the concentration of species X, i.e. the variable x, computes the mth root $a^* \mapsto x^* = \sqrt[m]{a^*}$ at speed that is at least

$$\begin{cases}\min\{\rho_a, 1\}, & \text{if } a^* \neq 0,\\ \min\left\{\frac{\rho_a}{m}, 1\right\}, & \text{if } a^* = 0.\end{cases}$$

Speed of an arbitrary arithmetic computation

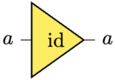
Theorem (Composite computations)

Consider a computation that is composed from a finite number of elementary computations.

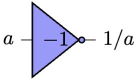
- *Then the speed of this composite computation is at least that of the slowest elementary computation.*
- *If none of the elementary computations is a root of zero, then the speed of the composite computation is at least 1.*

- Computation speed independent of number of elementary steps.
- Speed of each elementary step can be controlled.

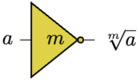
Elementary gates



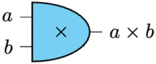
(a) Identification.



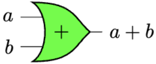
(b) Inversion.



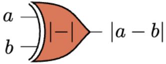
(c) mth root.



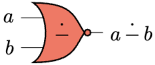
(d) Multiplication.



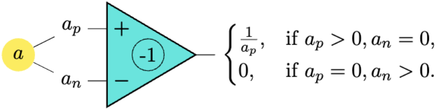
(e) Addition.



(f) Absolute difference.

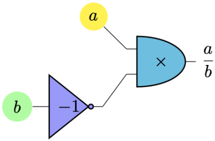


(g) Rectified subtraction.

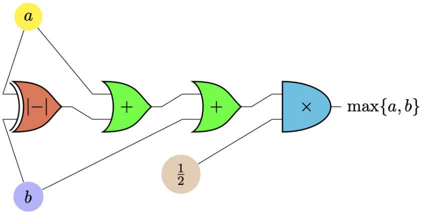


(h) Partial real inversion.

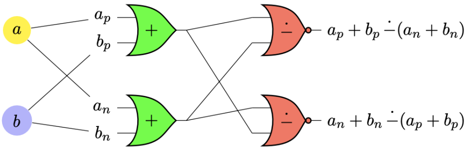
Division



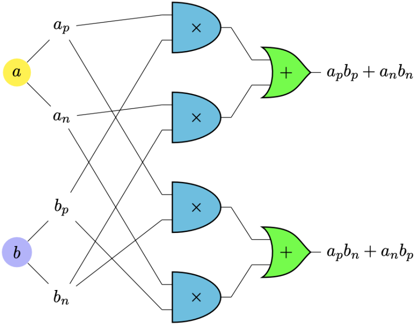
Maximum



Real Addition



Real Multiplication



Open questions

Proved results in *idealized* mass action setting.

- Arithmetic at input independent speed using only bimolecular reactions?
- When doing rectified subtraction $a \dot{-} b$, an intermediate variable goes to ∞ when $a = b$. Can this be avoided?
- Assumed all reactions have rate constant 1. What if rate constants are known, but not controllable? How should the constructions be modified?
- A root of zero is a bottleneck. Can this be avoided while keeping input-independent speed?
- Can the root of zero be used to advantage? Use it as "zero detector" or "equality detector"?
- What about power series or limiting processes?
- Can we compute other functions (eg. log, exp, sin etc.) at input independent speeds?
- Algorithms for Boolean and other algebras.

References:

- David F. Anderson, Badal Joshi, and Abhishek Deshpande. On reaction network implementations of neural networks, *Journal of Royal Society Interface*, Vol. 18, Issue 177, (April 2021), arXiv.
- David F. Anderson, and Badal Joshi. Chemical mass-action systems as analog computers: implementing arithmetic computations at specified speed, arXiv, submitted April 2024.

Acknowledgements:

- Work supported by NSF grant DMS-2051498.

Thank you!