Programming Molecules to Compute in Real Time

Xiang Huang

Feb 17, 2020

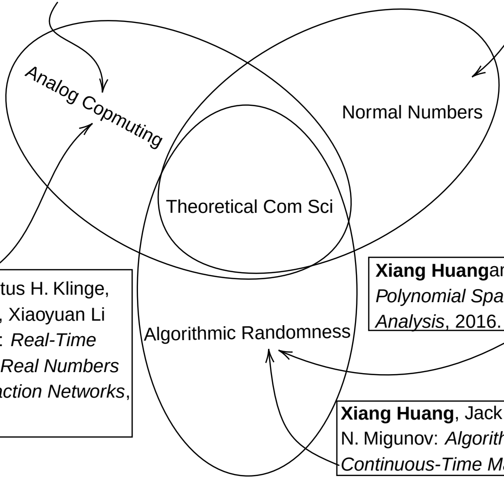
Computer Science Department, Iowa State University

Computer Science Department, Le Moyne College

Overall Research Topics

Xiang Huang, Titus H. Klinge, James I. Lathrop. *Real-Time Equivalence of Chemical Reaction Networks and Analog Computers*, 2019.

Xiang Huang, Jack H. Lutz, Elvira Mayordomo, and Donald M. Stull: *Asymptotic divergences and strong dichotomy, 2020*



Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li and Jack H. Lutz: *Real-Time Computability of Real Numbers by Chemical Reaction Networks*, 2017.

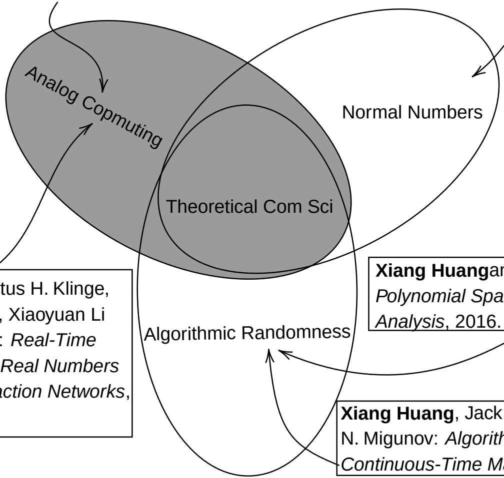
Xiang Huangand Donald. M. Stull: *Polynomial Space Randomness in Analysis*, 2016.

Xiang Huang, Jack H. Lutz, and Andrei N. Migunov: *Algorithmic Randomnessin Continuous-Time Markov Chains*, 2019.

Overall Research Topics

Xiang Huang, Titus H. Klinge, James I. Lathrop. *Real-Time Equivalence of Chemical Reaction Networks and Analog Computers*, 2019.

Xiang Huang, Jack H. Lutz, Elvira Mayordomo, and Donald M. Stull: *Asymptotic divergences and strong dichotomy, 2020*



Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li and Jack H. Lutz: *Real-Time Computability of Real Numbers by Chemical Reaction Networks*, 2017.

Xiang Huangand Donald. M. Stull: *Polynomial Space Randomness in Analysis*, 2016.

Xiang Huang, Jack H. Lutz, and Andrei N. Migunov: *Algorithmic Randomnessin Continuous-Time Markov Chains*, 2019.

Motivation & Introduction

Computability/Complexity of Real Numbers

Which (real) numbers can we compute?

Given a (new) computational model, this is a simple yet profound question to ask.

Which (real) numbers can we compute?

Given a (new) computational model, this is a simple yet profound question to ask.

- Turing (1936): computable real numbers on Turing Machines.

Computability/Complexity of Real Numbers

Which (real) numbers can we compute?

Given a (new) computational model, this is a simple yet profound question to ask.

- Turing (1936): computable real numbers on Turing Machines.
- Yamada (1962): Do that fast, please. (Real-time computability)

Computability/Complexity of Real Numbers

Which (real) numbers can we compute?

Given a (new) computational model, this is a simple yet profound question to ask.

- Turing (1936): computable real numbers on Turing Machines.
- Yamada (1962): Do that fast, please. (Real-time computability)
- Hartmanis & Stearns (1965): no can do, square root of two. (open conjecture)

Why We Care?

By answering questions like these, we can

Why We Care?

By answering questions like these, we can

- Showcase the power of our computational model.

Why We Care?

By answering questions like these, we can

- Showcase the power of our computational model.
- Guide algorithm design. Although the questions are purely theoretical, it is complicated enough to require sophisticated algorithm designs, which can be applied to more practical problems.

Why We Care?

By answering questions like these, we can

- Showcase the power of our computational model.
- Guide algorithm design. Although the questions are purely theoretical, it is complicated enough to require sophisticated algorithm designs, which can be applied to more practical problems.
- Better yet, give theoretical boundary of the computational model. (Identify "mission impossible".)

Why We Care?

By answering questions like these, we can

- Showcase the power of our computational model.
- Guide algorithm design. Although the questions are purely theoretical, it is complicated enough to require sophisticated algorithm designs, which can be applied to more practical problems.
- Better yet, give theoretical boundary of the computational model. (Identify "mission impossible".)

Today, we are going to advance our knowledge of the complexity of real numbers in the analog (CRNs and GPACs) models.

A CRN is an abstract **mathematical model** of how chemicals interact in a well-mixed volume.

CRNs and GPACs in a Nutshell

A CRN is an abstract **mathematical model** of how chemicals interact in a well-mixed volume.

Hydrazine Combustion

$$
\begin{aligned}
3N_2H_4 &\longrightarrow 4NH_3 + N_2 \\
N_2H_4 &\longrightarrow N_2 + 2H_2 \\
4NH_3 + N_2H_4 &\longrightarrow 3N_2 + 8H_2 \\
2H_2 + O_2 &\longrightarrow 2H_20
\end{aligned}
$$

A CRN is an abstract **mathematical model** of how chemicals interact in a well-mixed volume.

Hydrazine Combustion

$$3N_2H_4 \longrightarrow 4NH_3 + N_2$$
$$N_2H_4 \longrightarrow N_2 + 2H_2$$
$$4NH_3 + N_2H_4 \longrightarrow 3N_2 + 8H_2$$
$$2H_2 + O_2 \longrightarrow 2H_20$$

Abstract CRN

$$3A \longrightarrow 4B + C$$
$$A \longrightarrow C + 2D$$
$$4B + A \longrightarrow 3C + 8D$$
$$2D + E \longrightarrow 2F$$

CRNs and GPACs In A Nutshell

The General Purpose Analog Computer (GPAC) is an analog computer model first introduced in 1941 by Claude Shannon.

For simplicity, we can view both GPACs and CRNs as polynomial initial value problems (PIVPs):

GPAC/ODE

$$x_i{}' = p_i(x_1, \cdots, x_n),$$

for $i = 1, 2, \cdots, n$,

where p_i's are polynomials.

CRN/ODE[1]

$$x_i{}' = p_i(x_1, \cdots, x_n) - q_i(x_1, \cdots, x_n)\mathbf{x_i},$$

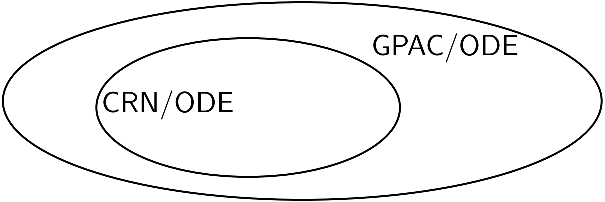
for $i = 1, 2, \cdots, n$,

where p_i's and q_i's are polynomials with **positive** coefficients.

[1] Vera Hars and János Tóth: On the Inverse Problem of Reaction Kinetics, 1979.

CRNs and GPACs

By the previous definition, CRNs are restricted GPACs. In terms of ODEs that they induce, we have



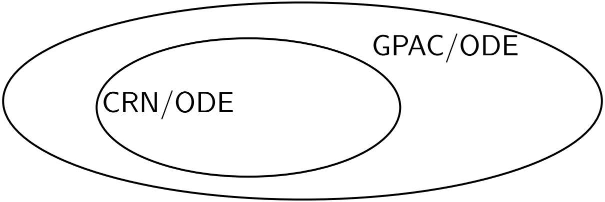
.

CRN:

$$x' = 1 - x.$$

CRNs and GPACs

By the previous definition, CRNs are restricted GPACs. In terms of ODEs that they induce, we have



.

CRN:

$$x' = 1 - x.$$

GPAC but not CRN:

$$x' = 1 - x,$$
$$y' = 1 - \boxed{x}.$$

Computable Real Numbers

Computable real numbers in **discrete** models:

1. In Turing's revolutionary 1936 paper, he defined a notion of computable real number.
   A number $\alpha \in \mathbb{R}$ is **computable**, if there exists a Turing machine (with no input) that prints the binary expansion of α to an output tape.

Computable Real Numbers

Computable real numbers in **discrete** models:

1. In Turing's revolutionary 1936 paper, he defined a notion of computable real number.
   A number $\alpha \in \mathbb{R}$ is **computable**, if there exists a Turing machine (with no input) that prints the binary expansion of α to an output tape.
2. Thus, the real number written on the output tape converges to α **in the limit**.

Computable Real Numbers

Computable real numbers in **discrete** models:

1. In Turing's revolutionary 1936 paper, he defined a notion of computable real number.
   A number $\alpha \in \mathbb{R}$ is **computable**, if there exists a Turing machine (with no input) that prints the binary expansion of α to an output tape.
2. Thus, the real number written on the output tape converges to α **in the limit**.

Inspired by the above notion, in analog models (GPAC/CRN), we can designate a variable x, such that

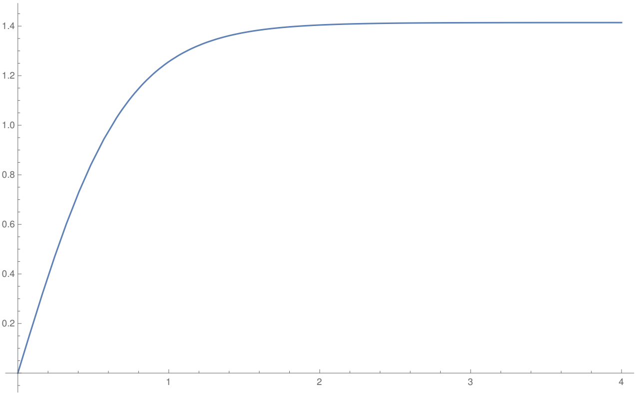
$$\lim_{t \to \infty} x(t) = \alpha.$$

And then we say x **computes** α.

GPAC/CRN computable numbers

Compute $\sqrt{2}$ in the limit, by the CRN/ODE:

$$x' = 2 - x^2$$



But how **fast** does it converge to $\sqrt{2}$?

Real-Time Computable Real-Numbers

In 1962, Yamada introduced a notion of computing a number $\alpha \in \mathbb{R}$ in **real time.**[2]

[2]Yamada, Hisao. "Real-time computation and recursive functions not real-time computable." IRE Transactions on Electronic Computers 6 (1962): 753-760.

[3]Lipton RJ (2017) Why the Hartmanis–Stearns conjecture is still open, 2012. Blog post. Retrieved 3 Feb from https://rjlipton.wordpress.com/2012/06/15/why-the-hartmanis-stearns-conjecture-is-still- open/

Real-Time Computable Real-Numbers

In 1962, Yamada introduced a notion of computing a number $\alpha \in \mathbb{R}$ in **real time.**[2]

Has been proven[3] to be equivalent to computing α in **linear time**, i.e., printing n bits takes $O(n)$-time. That is, getting an $O(2^{-n})$-approximation in $O(n)$-time.

[2] Yamada, Hisao. "Real-time computation and recursive functions not real-time computable." IRE Transactions on Electronic Computers 6 (1962): 753-760.

[3] Lipton RJ (2017) Why the Hartmanis–Stearns conjecture is still open, 2012. Blog post. Retrieved 3 Feb from https://rjlipton.wordpress.com/2012/06/15/why-the-hartmanis-stearns-conjecture-is-still- open/

Real-Time Computable Real-Numbers

In 1962, Yamada introduced a notion of computing a number $\alpha \in \mathbb{R}$ in **real time.**[2]

Has been proven[3] to be equivalent to computing α in **linear time**, i.e., printing n bits takes $O(n)$-time. That is, getting an $O(2^{-n})$-approximation in $O(n)$-time.

Similarly, in CRNs/GPACs, to define real-time computability, we can require our designated variable $x(t)$ such that

$$|x(t) - \alpha| < O(2^{-t}).$$

[2] Yamada, Hisao. "Real-time computation and recursive functions not real-time computable." IRE Transactions on Electronic Computers 6 (1962): 753-760.

[3] Lipton RJ (2017) Why the Hartmanis–Stearns conjecture is still open, 2012. Blog post. Retrieved 3 Feb from https://rjlipton.wordpress.com/2012/06/15/why-the-hartmanis-stearns-conjecture-is-still- open/

Real-Time CRN Computable Reals

$\alpha \in \mathbb{R}$ is **real-time CRN-computable** (resp., **GPAC-computable**), and we write $\alpha \in \mathbb{R}_{RTCRN}$ (resp., $\alpha \in \mathbb{R}_{RTGPAC}$), if there exist an (**integral**) PIVP induced by a CRN (resp. GPAC) and a variable $x(t)$ in the system with the following properties:

1. (**zero initial values**). For all variable $y(t)$, $y(0) = 0$. (This can be relaxed to integral values.)

Real-Time CRN Computable Reals

$\alpha \in \mathbb{R}$ is **real-time CRN-computable** (resp., **GPAC-computable**), and we write $\alpha \in \mathbb{R}_{RTCRN}$ (resp., $\alpha \in \mathbb{R}_{RTGPAC}$), if there exist an (**integral**) PIVP induced by a CRN (resp. GPAC) and a variable $x(t)$ in the system with the following properties:

1. (**zero initial values**). For all variable $y(t)$, $y(0) = 0$. (This can be relaxed to integral values.)
2. (**boundedness**). All variables are bounded by some constant M,
$$y(t) \leq M.$$

Real-Time CRN Computable Reals

$\alpha \in \mathbb{R}$ is **real-time CRN-computable** (resp., **GPAC-computable**), and we write $\alpha \in \mathbb{R}_{RTCRN}$ (resp., $\alpha \in \mathbb{R}_{RTGPAC}$), if there exist an (**integral**) PIVP induced by a CRN (resp. GPAC) and a variable $x(t)$ in the system with the following properties:

1. (**zero initial values**). For all variable $y(t)$, $y(0) = 0$. (This can be relaxed to integral values.)
2. (**boundedness**). All variables are bounded by some constant M,
$$y(t) \leq M.$$
3. (**real-time convergence**).
$$|x(t) - |\alpha|| \leq O(2^{-t}).$$

We[4] showed that

Theorem
All algebraic real numbers are in $\mathbb{R}_{RTCRN}$.

[4] Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li, and Jack H. Lutz. "Real-time computability of real numbers by chemical reaction networks." Natural Computing 18.1 (2019): 63-73.

Basic Results

We[4] showed that

Theorem
All algebraic real numbers are in $\mathbb{R}_{RTCRN}$.

Theorem
Some transcendental numbers are in $\mathbb{R}_{RTCRN}$.

[4] Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li, and Jack H. Lutz. "Real-time computability of real numbers by chemical reaction networks." Natural Computing 18.1 (2019): 63-73.

Basic Results

We[4] showed that

Theorem
All algebraic real numbers are in $\mathbb{R}_{RTCRN}$.

Theorem
Some transcendental numbers are in $\mathbb{R}_{RTCRN}$.

- Two (ugly) numbers u, v, such that $u - v = e$.

[4] Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li, and Jack H. Lutz. "Real-time computability of real numbers by chemical reaction networks." Natural Computing 18.1 (2019): 63-73.

We[4] showed that

Theorem
All algebraic real numbers are in $\mathbb{R}_{RTCRN}$.

Theorem
Some transcendental numbers are in $\mathbb{R}_{RTCRN}$.

- Two (ugly) numbers u, v, such that $u - v = e$.
- Did not know how to prove **subtraction** can be done in **real time**.

[4] Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li, and Jack H. Lutz. "Real-time computability of real numbers by chemical reaction networks." Natural Computing 18.1 (2019): 63-73.

Basic Results

We[4] showed that

Theorem
All algebraic real numbers are in $\mathbb{R}_{RTCRN}$.

Theorem
Some transcendental numbers are in $\mathbb{R}_{RTCRN}$.

- Two (ugly) numbers u,v, such that $u - v = e$.
- Did not know how to prove **subtraction** can be done in **real time**.
- Did not know $\mathbb{R}_{RTCRN}$ is a field. Hard to compute real numbers in general because of lack of field structure.

[4] Xiang Huang, Titus H. Klinge, James I. Lathrop, Xiaoyuan Li, and Jack H. Lutz. "Real-time computability of real numbers by chemical reaction networks." Natural Computing 18.1 (2019): 63-73.

Newer Results

In the newer paper[5], we have

1. $\mathbb{R}_{RTCRN}$ is a field. In particular, **subtraction** can be done in real time.

[5] Xiang Huang, Titus H. Klinge, James I. Lathrop. Real-Time Equivalence of Chemical Reaction Networks and Analog Computers. DNA 2019.

Newer Results

In the newer paper[5], we have

1. $\mathbb{R}_{RTCRN}$ is a field. In particular, **subtraction** can be done in real time.
2. $\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$.

[5] Xiang Huang, Titus H. Klinge, James I. Lathrop. Real-Time Equivalence of Chemical Reaction Networks and Analog Computers. DNA 2019.

Newer Results

In the newer paper[5], we have

1. $\mathbb{R}_{RTCRN}$ is a field. In particular, **subtraction** can be done in real time.
2. $\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$.
3. $\mathbb{R}_{RTCRN}$ contains the transcendental numbers e and π.

[5] Xiang Huang, Titus H. Klinge, James I. Lathrop. Real-Time Equivalence of Chemical Reaction Networks and Analog Computers. DNA 2019.

Implications

1. $\mathbb{R}_{RTCRN}$ is a field:
   - modularize/reuse your CRN algorithms.

Implications

1. $\mathbb{R}_{RTCRN}$ is a field:
    - modularize/reuse your CRN algorithms.
2. $\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$:
    - One can implement an algorithm by a GPAC, which is (much) simpler in form. Then compile it into a CRN.
    - It greatly simplifies the proof that a number $\alpha \in \mathbb{R}_{RTCRN}$.

Implications

1. $\mathbb{R}_{RTCRN}$ is a field:
   - modularize/reuse your CRN algorithms.
2. $\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$:
   - One can implement an algorithm by a GPAC, which is (much) simpler in form. Then compile it into a CRN.
   - It greatly simplifies the proof that a number $\alpha \in \mathbb{R}_{RTCRN}$.
3. e, $\pi \in \mathbb{R}_{RTCRN}$: the field $\mathbb{R}_{RTCRN}$ is not boring.

$\mathbb{R}_{RTCRN}$ is a Field

Closure Under Addition, Multiplication

Lemma (Addition)
If $\alpha, \beta \in \mathbb{R}_{RTCRN}$*, then* $\alpha + \beta \in \mathbb{R}_{RTCRN}$.

Lemma (Multiplication)
If $\alpha, \beta \in \mathbb{R}_{RTCRN}$*, then* $\alpha\beta \in \mathbb{R}_{RTCRN}$.

Subtraction Is Hard

- Assume α, β can be computed by CRN in real time, we hope that we can compute $\alpha - \beta$ in real time, too.

[6] Buisman, H.J. et al.: Computing algebraic functions with biochemical reaction networks. Artificial Life pp. 5–19 (2009)

Subtraction Is Hard

- Assume α, β can be computed by CRN in real time, we hope that we can compute $\alpha - \beta$ in real time, too.
- There are known subtraction algorithms[6], but we do not know how to prove these algorithms converge exponentially fast.

[6] Buisman, H.J. et al.: Computing algebraic functions with biochemical reaction networks. Artificial Life pp. 5–19 (2009)

Subtraction Is Hard

- Assume α, β can be computed by CRN in real time, we hope that we can compute $\alpha - \beta$ in real time, too.
- There are known subtraction algorithms[6], but we do not know how to prove these algorithms converge exponentially fast.
- We developed a subtraction algorithm that we can **prove** that it converges exponentially fast.

[6] Buisman, H.J. et al.: Computing algebraic functions with biochemical reaction networks. Artificial Life pp. 5–19 (2009)

A Convergence Lemma

Let $f(t)$ be a function that converges to $\alpha \in \mathbb{R}$ **exponentially fast**, that is, there exist constants τ and γ such that for all $t \in [\tau, \infty)$

$$|f(t) - \alpha| \leq e^{-\gamma t}.$$

Lemma (Reciprocal Convergence)
If $x(t)$ is a function that satisfies

$$x' = 1 - f(t) \cdot x,$$

then $x(t)$ converges to $\frac{1}{\alpha}$ exponentially fast.

Lemma (Reciprocal)
If $\alpha \in \mathbb{R}_{RTCRN}$, then $\frac{1}{\alpha} \in \mathbb{R}_{RTCRN}$.

Subtraction: Two Reciprocals for One Subtraction

Lemma (Subtraction)
If $\alpha, \beta \in \mathbb{R}_{RTCRN}$ *and* $\alpha > \beta > 0$, *then* $\alpha - \beta \in \mathbb{R}_{RTCRN}$.

Proof Sketch
Let x, y computes α, β in real time.

1. **reciprocal:** Introduce a variable r

$$r' = 1 - (x - y)r.$$

2. **reciprocal of reciprocal:** Introduce a variable z

$$z' = 1 - r \cdot z.$$

- r, z are **solvable** and easy to **analyze**.
- r, z are CRN-implementable.
- z converges to $\alpha - \beta$ in real-time.

$\mathbb{R}_{RTCRN}$ is a Field

Theorem
$\mathbb{R}_{RTCRN}$ is a field.

Proof.
This follows immediately by closure under addition, subtraction, multiplication, and reciprocal. $\square$

$$\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$$

Proof that $\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$

Theorem
$\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$

Theorem
$\mathbb{R}_{RTCRN} = \mathbb{R}_{RTGPAC}$

Proof ($\subseteq$).
CRNs are special forms of GPACs. □

The backward direction is the non-trivial direction. We need to use a **souped-up version** of the following theorem.

Theorem (Difference Encoding[7])
Given a GPAC $\mathcal{A}$, there is a CRN $\mathcal{N}$ that for each variable x in $\mathcal{A}$, there are two variables x_1 and x_2 in $\mathcal{N}$ such that

$$x(t) = x_1(t) - x_2(t) \quad \textbf{pointwise for every t} \geq \mathbf{0}.$$

This means we can compile a GPAC into a CRN.

We use a similar construction. Moreover, we also show that **if $x(t)$ is bounded, then x_1 and x_2 are also bounded**.

[7] François Fages, Guillaume Le Guludec, Olivier Bournez, and Amaury Pouly. "Strong turing completeness of continuous chemical reaction networks and compilation of mixed analog-digital programs." International Conference on Computational Methods in Systems Biology. Springer, Cham, 2017.

Example: Difference Encoding

The function $v(t) = \sin(t)$ can be implemented by the top-left GPAC/ODE, and encoded by $v_1(t) - v_2(t)$ in the bottom-left CRN/ODE.

GPAC/ODE

$$u' = v$$

$$v' = 1 - u$$

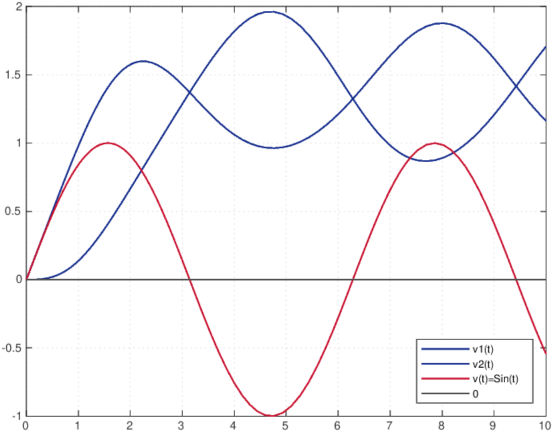
CRN/ODE

$$u_1' = v_1 - u_1 u_2$$

$$u_2' = v_2 - u_1 u_2$$

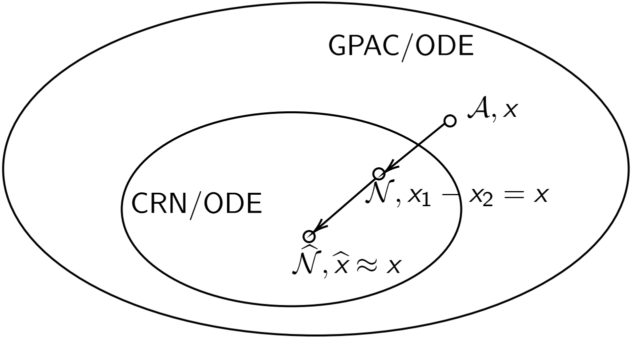
$$v_1' = 1 + u_2 - v_1 v_2$$

$$v_2' = u_1 - v_1 v_2$$



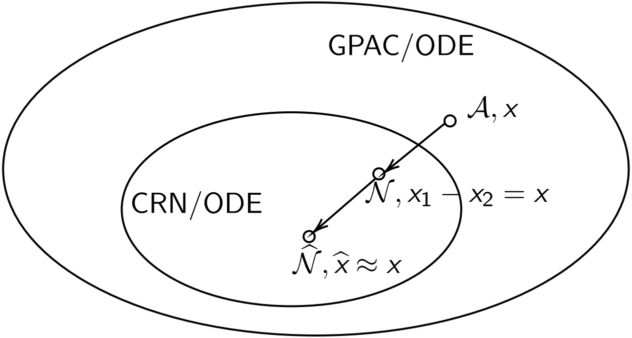
Plot: $\sin(t) = v_1(t) - v_2(t)$

The Rest of The Proof



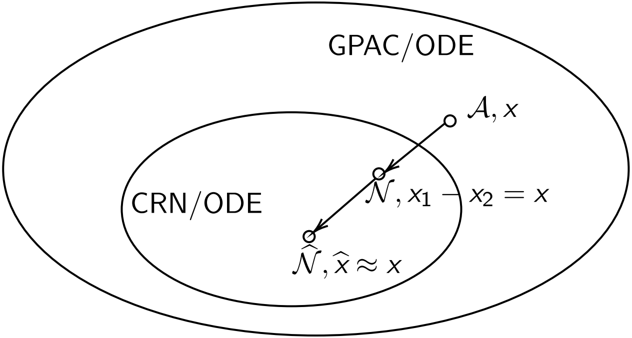
- Let x in GPAC $\mathcal{A}$ computes α in real time.

The Rest of The Proof



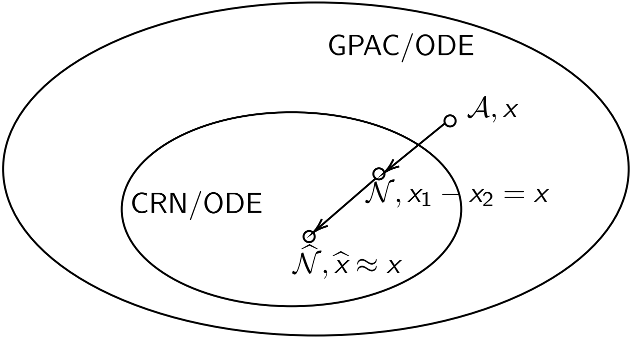
- Let x in GPAC $\mathcal{A}$ computes α in real time.
- Now the CRN $\mathcal{N}$ encodes x by **two** variables $x_1 - x_2$.

The Rest of The Proof



- Let x in GPAC $\mathcal{A}$ computes α in real time.
- Now the CRN $\mathcal{N}$ encodes x by **two** variables $x_1 - x_2$.
- Can we do it in **one single variable**?
  - Yes. By two applications of the reciprocal lemma, we get $\widehat{\mathcal{N}}$.

The Rest of The Proof



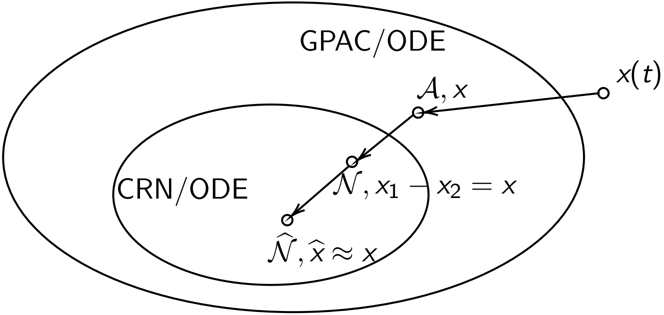
- Let x in GPAC $\mathcal{A}$ computes α in real time.
- Now the CRN $\mathcal{N}$ encodes x by **two** variables $x_1 - x_2$.
- Can we do it in **one single variable**?
  - Yes. By two applications of the reciprocal lemma, we get $\widehat{\mathcal{N}}$.
- By $\widehat{x} \approx x$, we mean it computes the same α in real time.

e and π are in $\mathbb{R}_{RTCRN}$

A General Procedure

To Compute a number α in real time by CRN, we do

1. Pick a function $x(t)$ that converges to α **exponentially** fast. (creative)
2. Implement $x(t)$ by a GPAC. (creative)
3. Translate the GPAC into a CRN $\mathcal{N}$. (automatic)
4. Lastly, turn $\mathcal{N}$ into $\widehat{\mathcal{N}}$. (automatic)



π is in $\mathbb{R}_{RTCRN}$

Theorem

$\pi \in \mathbb{R}_{RTCRN}$.

Theorem
$\pi \in \mathbb{R}_{RTCRN}$.

As a first attempt, we can use the fact that

$$\lim_{t \to \infty} \arctan(t) = \frac{\pi}{2}.$$

Let

$$x(t) = \arctan(t), \qquad y(t) = \frac{1}{1+t^2}, \qquad z(t) = \frac{t}{1+t^2}.$$

Theorem
$\pi \in \mathbb{R}_{RTCRN}$.

As a first attempt, we can use the fact that

$$\lim_{t\to\infty} \arctan(t) = \frac{\pi}{2}.$$

Let

$$x(t) = \arctan(t), \qquad y(t) = \frac{1}{1+t^2}, \qquad z(t) = \frac{t}{1+t^2}.$$

Then we have

$$\begin{aligned} x' &= y \\ y' &= -2yz, \\ z' &= y^2 - z^2. \end{aligned}$$

with initial value $x(0) = z(0) = 0$ and $y(0) = 1$.

π is in $\mathbb{R}_{RTCRN}$

Unfortunately, $\arctan(t)$ does not converge to $\frac{\pi}{2}$ fast enough.

At the second attempt, we pick the function

$$x(t) = \arctan(1 - e^{-t}).$$

Note that $x(t)$ converges to $\frac{\pi}{4}$ exponentially fast.

We skip the rest of the construction and proof here.

Then the following ODE system can be implemented by a CRN.

$$u_1' = p_x^+ - u_1 v_1 (p_x^+ + p_x^-), \tag{46}$$
$$v_1' = p_x^- - u_1 v_1 (p_x^+ + p_x^-), \tag{47}$$
$$u_2' = p_2^+ - u_2 v_2 (p_2^+ + p_2^-), \tag{48}$$
$$v_2' = p_2^- - u_2 v_2 (p_2^+ + p_2^-), \tag{49}$$
$$u_3' = p_3^+ - u_3 v_3 (p_3^+ + p_3^-), \tag{50}$$
$$v_3' = p_3^- - u_3 v_3 (p_3^+ + p_3^-), \tag{51}$$
$$u_4' = p_4^+ - u_4 v_4 (p_4^+ + p_4^-), \tag{52}$$
$$v_4' = p_4^- - u_4 v_4 (p_4^+ + p_4^-), \tag{53}$$

where all variables are initialized to be zero.

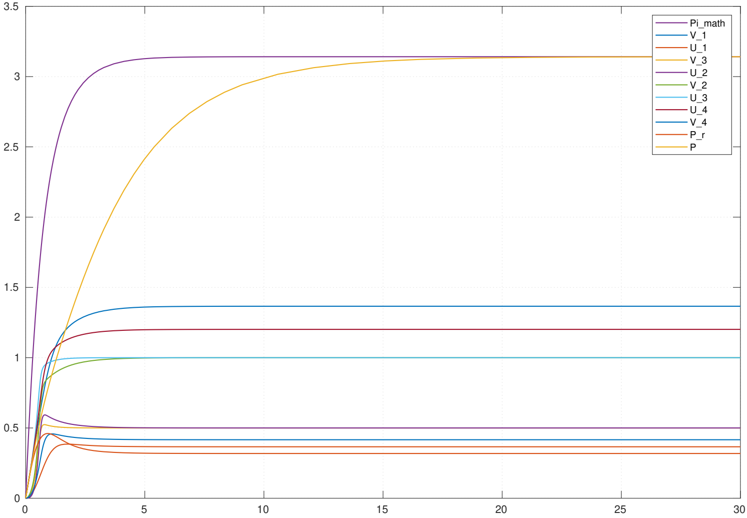
$$\begin{aligned} p_3^+ &= u_1(u_2+1)^2 + u_1 v_2^2 + 2v_1 v_2(u_2+1) + v_1 u_3^2 + v_1 v_3^2 \\ &\quad + 2u_1 u_3 v_3 + (u_2+1)^2 + v_2^2 + 2u_3 v_3 \\ p_3^- &= v_1(u_2+1)^2 + v_1 v_2^2 + 2u_1(u_2+1)v_2 + u_1 u_3^2 + u_1 v_3^2 \\ &\quad + 2v_1 u_3 v_3 + 2(1+u_2)v_2 + u_3^2 + v_3^2 \end{aligned}$$

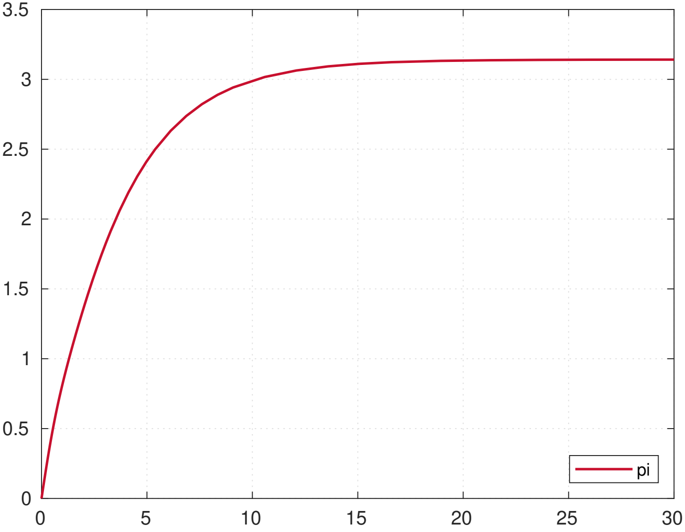
Reactions resulting from $-u_3v_3p_3^+$:

$$U_1 + U_3 + V_3 \xrightarrow{1} U_1, \qquad U_1 + U_2 + U_3 + V_3 \xrightarrow{2} U_1 + U_2,$$
$$U_1 + 2U_2 + U_3 + V_3 \xrightarrow{1} U_1 + 2U_2, \qquad U_1 + 2V_2 + U_3 + V_3 \xrightarrow{1} U_1 + 2V_2,$$
$$V_1 + V_2 + U_3 + V_3 \xrightarrow{2} V_1 + V_2, \qquad V_1 + U_2 + V_2 + U_3 + V_3 \xrightarrow{2} V_1 + U_2 + V_2,$$
$$V_1 + 3U_3 + V_3 \xrightarrow{1} V_1 + 2U_3, \qquad V_1 + U_3 + 3V_3 \xrightarrow{1} V_1 + 2V_3,$$
$$U_1 + 2U_3 + 2V_3 \xrightarrow{2} U_1 + U_3 + V_3, \qquad 2U_2 + U_3 + V_3 \xrightarrow{1} 2U_2,$$
$$U_2 + U_3 + V_3 \xrightarrow{2} U_2, \qquad U_3 + V_3 \xrightarrow{1} \emptyset,$$
$$2V_2 + U_3 + V_3 \xrightarrow{1} 2V_2, \qquad 2U_3 + 2V_3 \xrightarrow{2} U_3 + V_3.$$

Reactions resulting from $-u_3v_3p_3^-$:

$$V_1 + U_3 + V_3 \xrightarrow{1} V_1, \qquad V_1 + U_2 + U_3 + V_3 \xrightarrow{2} U_1 + V_1,$$
$$V_1 + 2U_2 + U_3 + V_3 \xrightarrow{1} V_1 + 2U_2, \qquad V_1 + 2V_2 + U_3 + V_3 \xrightarrow{1} V_1 + 2V_2,$$
$$U_1 + V_2 + U_3 + V_3 \xrightarrow{2} U_1 + V_2, \qquad U_1 + U_2 + V_2 + U_3 + V_3 \xrightarrow{2} U_1 + U_2 + V_2,$$
$$U_1 + 3U_3 + V_3 \xrightarrow{1} U_1 + 2U_3, \qquad U_1 + U_3 + 3V_3 \xrightarrow{1} U_1 + 2V_3,$$
$$V_1 + 2U_3 + 2V_3 \xrightarrow{2} V_1 + U_3 + V_3, \qquad V_2 + U_3 + V_3 \xrightarrow{2} V_2,$$
$$U_2 + V_2 + U_3 + V_3 \xrightarrow{2} U_2 + V_2, \qquad 3U_3 + V_3 \xrightarrow{1} 2U_3,$$
$$U_3 + 3V_3 \xrightarrow{1} 2V_3.$$


Pi_math
V_1
U_1
V_3
U_2
V_2
U_3
U_4
V_4
P_r
P
0
0.5
1
1.5
2
2.5
3
3.5
0
5
10
15
20
25
30



Plot of π by MATLAB/Simbiology. (CRN size: 10 species and 120 reactions.)

Conclusion

1. We manipulate (abstract) molecules to compute real numbers in real time.
   - A good way to learn ODE.
   - In principle, you can use DNA strands to implement the abstract reactions.
2. Computability and complexity in analog models brings new challenges and opportunities.
3. Interdisciplinary studies: descriptive set theory, algebraic geometric, nonlinear systems, DNA/molecular computing, wet lab implementation ...

Thank you!