the limits of dynamical systems

DS represent agents only in terms of the interactions that exist and are assumed relevant in a given context; they do not represent agents in terms of potential interactions that are not causally effective now, but would become so if context was changing.

To describe agents in these terms requires a representation of their structure, which holds the key to possible interactions that can become actualized in other contexts.

It is an open question to what extent physical structure and modes of interaction can be lifted into a formal logical representation that permits such reasoning in the molecular realm.

*The bottleneck in biology is not quantification, but description.* (ouch!)
soup of lambda-expressions
(all terms closed and in normal form)
Self-maintenance is the consequence of a constructive feedback loop: it occurs when the construction processes induced by the constituents of a system permit the continuous regeneration of these same constituents.

Immanuel Kant (Kritik der Urteilskraft, 1790): “[...] an organized product of nature is one in which all is end and, reciprocally, means too.”
parse all terms into prefixes (not a $\lambda$-term) and terminals (closed $\lambda$-term)

self-maintaining ensemble

derive the transformational behavior of building blocks and describe all interactions in terms of a set of rewrite rules

yields a specific algebraic structure

all reference to the underlying micro-mechanics ($\lambda$-calculus) has been removed
perturbation

constrained extension

perturbation
Applying a function to an argument in a typed calculus ...

\[
\begin{array}{c}
A \\
\rightarrow \\
B \\
\hline
A \\
\end{array}
\]

... corresponds to *modus ponens* in logic

This correspondence is known as the Curry-Howard isomorphism:

- type $\sigma$ \leftrightarrow logical formula (predicate) $\sigma$
- $\lambda$-term of type $\sigma$ \leftrightarrow proof of $\sigma$

so, our picture of chemistry becomes....

<table>
<thead>
<tr>
<th>molecular shape</th>
<th>chemical predicate $\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>molecule with that shape</td>
<td>proof of the validity of $\sigma$</td>
</tr>
<tr>
<td>chemical reaction</td>
<td>rule of inference</td>
</tr>
<tr>
<td>reactants / products</td>
<td>(proofs of) premises / conclusions</td>
</tr>
</tbody>
</table>
A more formal approach to agent-based systems is the \( \pi \)-calculus. Its application to biology was pioneered by Shapiro, Regev, and Priami. Its potential for biology was considerably deepened by Danos and Laneve.

The \( \pi \)-calculus
(Milner, Walker and Parrow 1989)

- a program specifies a network of interacting processes
- processes are defined by their potential communication activities
- communication occurs on complementary channels, identified by names
- message content: channel name
Processes and channels

\( P, Q, \cdots \) process names (2.1)

\( x, y, \cdots \) channel names (2.2)

Events

\( \overline{x}, \overline{y}, \cdots \) channel co-names (2.3)

\( \pi ::= x \) communication on channel name \( x \) (2.4)

\( \overline{x} \) communication on channel co-name \( x \) (2.5)

\( x(y) \) receive \( y \) along \( x \) (2.6)

\( \overline{x}(y) \) send \( y \) along \( x \) (2.7)

Process syntax

\( P ::= P_1 | P_2 \) parallel processes (2.8)

\( \pi . P_1 \) sequential prefixing by communication (2.9)

\( \pi_1 . P_1 + \pi_2 . P_2 \) mutually exclusive communications (2.10)

\( (\text{new } x)P \) new communication scope (2.11)

\( 0 \) inert process (2.12)

Structural congruence

\( P | Q \equiv Q | P \) commutativity of PAR (2.13)

\( (P | Q) | R \equiv P | (Q | R) \) associativity of PAR (2.14)

\( P + Q \equiv Q + P \) commutativity of summation (2.15)

\( P + Q + R \equiv P + (Q + R) \) associativity of summation (2.16)

\( (\text{new } x)0 \equiv 0 \) scope of inert processes (2.17)

\( (\text{new } x)(\text{new } y)P \equiv (\text{new } y)(\text{new } x)P \) multiple communication scopes (2.18)

\( ((\text{new } x)P) | Q \equiv (\text{new } x)(P | Q) \) if \( x \notin FN(Q) \) scope extrusion (2.19)

\( A(\overline{y}) \equiv \{\overline{y}/\overline{x}\}Q_A \) recursive parametric definition (2.20)

\( x(y).P = x(z).(\{z/y\}P) \) if \( z \notin FN(P) \) renaming of input channel \( y \) (2.21)

\( (\text{new } y).P = (\text{new } z).(\{z/y\}P) \) if \( z \notin FN(P) \) renaming of restricted channel \( y \) (2.22)

Reaction rules

\( (\cdots + \overline{x}(z).Q)(\cdots + x(y).P) \rightarrow Q|P\{z/y\} \) communication (COMM) (2.23)

if \( P \rightarrow P' \) then \( P|Q \rightarrow P'|Q \) reaction under parallel composition (2.24)

if \( P \rightarrow P' \) then \( (\text{new } x)P \rightarrow (\text{new } x)P' \) reaction within restricted scope (2.25)

if \( Q \equiv P, P \rightarrow P' \), and \( P' \equiv Q' \) then \( Q \rightarrow Q' \) reaction up to structural congruence (2.26)
E + S ⇄ ES → E + P

- Says nothing about internal structure of E, S, P, ES
- We want to encode the reaction scheme... but according to certain principles

\[
\begin{align*}
\ll [ E + S & \leftrightarrow ES \rightarrow E + P ]_\pi \\
& \ll [- \rightarrow_{\text{CHEM}} - ]_\pi = - \rightarrow^* - \\
& \ll [- +_{\text{CHEM}} - ]_\pi = - | - 
\end{align*}
\]

L. Greg Meredith (2005)
from these we deduce

- \([E + S]_{\pi} = [E]_{\pi} \mid [S]_{\pi} \rightarrow^{*} [ES]_{\pi}\)
- \([ES]_{\pi} \rightarrow^{*} ( [E]_{\pi} \mid [S]_{\pi} ) + ( [E]_{\pi} \mid [P]_{\pi} )\)

from these we deduce

- \(\exists x_{0}. ( [E]_{\pi} \approx (\nu \ e)(x_{0}[e].[E]_{\pi}^{'} + X_{E}) ) \land ( [S]_{\pi} \approx x_{0}(y).[S]_{\pi}^{'} + X_{S} )\)
- \([ES]_{\pi} \approx (\nu \ e)([E]_{\pi}^{'} \mid [S]_{\pi}^{'}{e/y})\)

therefore

- \((\nu \ e)([E]_{\pi}^{'} \mid [S]_{\pi}^{'}{e/y}) \rightarrow^{*} ( [E]_{\pi} \mid [S]_{\pi} ) + ( [E]_{\pi} \mid [P]_{\pi} )\)

L.Greg Meredith (2005)
since $E$ is an enzyme, $\llbracket E \rrbracket_\pi$ is the future of $\llbracket E \rrbracket_\pi'$, and $\llbracket S \rrbracket_\pi$ and $\llbracket P \rrbracket_\pi$ are the futures of $\llbracket S \rrbracket_\pi'{e/y}$

- $(\forall e)(\llbracket E \rrbracket_\pi' \mid \llbracket S \rrbracket_\pi'{e/y}) \rightarrow^* (\llbracket E \rrbracket_\pi \mid \llbracket S \rrbracket_\pi) + (\llbracket E \rrbracket_\pi \mid \llbracket P \rrbracket_\pi)$

implies

- $\exists x_1 x_2. (\llbracket E \rrbracket_\pi' = x_1(y).\llbracket E \rrbracket_\pi + x_2(y).\llbracket E \rrbracket_\pi + X_E) \& (\llbracket S \rrbracket_\pi' = x_1[e].\llbracket S \rrbracket_\pi + x_2[e].\llbracket P \rrbracket_\pi + X_S)$

setting $X$'s to $0$ and minimizing the number of $\rightarrow^\pi$ steps we arrive at

- $\llbracket E \rrbracket_\pi = (\forall e)(x_0[e].(x_1(y).\llbracket E \rrbracket_\pi + x_2(y).\llbracket E \rrbracket_\pi))$
- $\llbracket S \rrbracket_\pi = x_0(y).(x_1[e].\llbracket S \rrbracket_\pi + x_2[e].\llbracket P \rrbracket_\pi)$

L.Greg Meredith (2005)
use spatial logic (L.Caires) to capture the logical content (the characteristic formula F) of the process corresponding to this reaction

translate biological networks into pi-processes $x_i$

model-check F against the $x_i$

thus identify networks with a (possibly dynamic) communication structure that behave like F (have that type)

$$E + S \leftrightarrow ES \rightarrow E + P$$

is really a reaction (network) type.
The logic formula, “the largest process X that behaves in some way and eventually becomes X”, describes the type “catalyst”, which picks out the following red processes:
The spatial logic formula, "the largest process X that behaves in some way and eventually becomes X|X", describes the type "autocatalyst", which picks out the following red processes:
an autocatalytic network at the dawn of life?

The search for networks that inhabit certain types is important, because it extends current efforts at detecting network motifs.

Such efforts focus on syntactical motifs, but network types are behavioral motifs!
• detect whether, in a network, certain subgraphs occur more frequently than expected (expectation means a suitably randomized control)
• those that do are presumably solutions to some problem(s)
• figure out the problem(s)

example:
feed-forward loop

the motives of motifs: “feed forward loop”

a delay mechanism...

...implementing a pulse-filter
are these types expressible and checkable?  
is “delay” a type?

feed-forward

“delay” to filter

multisite phosphorylation

“delay” to postpone commitment

kinetic proofreading

“delay” to postpone commitment
are these types expressible and checkable?
from a physics of information to a biology of information
got guts?

i’m looking for a postdoc at the concurrency/biology interface of type:

must survive in a lab atmosphere & talk to biologists & have some physics intuition.

(is this type inhabited?)