Our understanding of life will derive in large measure from how we understand chemistry.
Walter Fontana and Leo W. Buss, 1996
From Chemistry to ODEs
Chemical Reactions

\[ A \rightarrow^r B_1 + \ldots + B_n \] Degradation \[ [A]^* = -r[A] \] \[ A_1 + A_2 \rightarrow^r B_1 + \ldots + B_n \] Asymmetric Collision \[ [A_i]^* = -r[A_1][A_2] \] \[ A + A \rightarrow^r B_1 + \ldots + B_n \] Symmetric Collision \[ [A]^* = -r[A][A]-1 \] (assuming \( A \neq B_i \neq A_j \) for all \( i,j \))

No other reactions!

Trimolecular reactions:
\[ A + B + C \rightarrow^r D \]

the measured “\( r \)” is an (imperfect) aggregate of e.g.:
\[ A + B \rightleftharpoons r^1 AB \]
\[ AB + C \rightarrow r^3 D \]

Enzymatic reactions:
\[ S \stackrel{E}{\rightarrow}^r P \]

the “\( r \)” is given by Michaelis-Menten (approximated steady-state) laws:
\[ E + S \rightleftharpoons r^1 ES \]
\[ ES \rightarrow r^3 P + E \]
From Reactions to ODE's

\[ v_1: A+B \rightarrow k_1 C+C \]
\[ v_2: A+C \rightarrow k_2 D \]
\[ v_3: C \rightarrow k_3 E+F \]
\[ v_4: F+F \rightarrow k_4 B \]

Set a rate law for each reaction (Degradation/Asymmetric/Symmetric)

\[ \text{Rate laws} \]
\[ \text{Stoichiometric matrix} \]
\[ \text{Quantity changes} \]

\[ \mathbf{X}^\bullet = \mathbf{N} \cdot \mathbf{l} \]

\[ \mathbf{[A]}^\bullet = -l_1 - l_2 \]
\[ \mathbf{[B]}^\bullet = -l_1 + l_4 \]
\[ \mathbf{[C]}^\bullet = 2l_1 - l_2 - l_3 \]
\[ \mathbf{[D]}^\bullet = l_2 \]
\[ \mathbf{[E]}^\bullet = l_3 \]
\[ \mathbf{[F]}^\bullet = l_3 - 2l_4 \]

\[ \begin{array}{cccc}
A & v_1 & v_2 & v_3 & v_4 \\
B & 1 & 1 & 1 & 1 \\
C & 2 & 1 & 1 & 1 \\
D & 1 & 1 & 1 & 1 \\
E & 1 & 1 & 1 & 1 \\
F & 1 & 1 & 1 & 1 \\
\end{array} \]

\[ \begin{array}{c}
l_1 \\
l_2 \\
l_3 \\
l_4 \\
\end{array} \]

\[ \begin{array}{c}
k_1[A][B] \\
k_2[A][C] \\
k_3[C] \\
k_4[F](F-1)/2 \\
\end{array} \]

---

\[ \begin{array}{c}
x \text{chemical species} \\
[-]: \text{quantity of molecules} \\
l: \text{rate laws} \\
k: \text{kinetic parameters} \\
N: \text{stoichiometric matrix} \\
\end{array} \]
The proper rate law $l_4$ of the symmetric reaction is $k_4[F][(F)-1]/2$, because that is the number of possible collisions between $[F]$ particles.

But the $v_4$ contribution to $[F]^\cdot$ is $-k_4[F][(F)-1]$ because 2 $F$ are consumed in that reaction!

Compare with the contribution of $v_4$ to $[B]^\cdot$, which is $k_4[F][(F)-1]/2$.

That’s why we said earlier that

\[ A + A \rightarrow^r B_1 + \ldots + B_n \]

gives $[A]^\cdot = -r[A][(A)-1]$ (2 $A$ consumed)

Instead, e.g.:

\[ A + A \rightarrow^r A + B \]

gives $[A]^\cdot = -r[A][(A)-1]/2$ (1 $A$ consumed)
From Chemistry to Processes


### Chemical Ground Form (CGF)

- **Definitions** \((n \geq 0)\)
  \[ E ::= X_1 = M_1, \ldots, X_n = M_n \]
- **Molecules** \((n \geq 0)\)
  \[ M ::= \pi_1; P_1 \oplus \ldots \oplus \pi_n; P_n \]
- **Solutions** \((n \geq 0)\)
  \[ P ::= X_1 \mid \ldots \mid X_n \]
- **Interactions** (delay, input, output)
  \[ \pi ::= \tau_r \ ?n(r) \!n(r) \]
- **Definitions with Initial Conditions**
  \[ CGF ::= E, P \]

\( \oplus \) is stochastic choice (vs. \(+\) for chemical reactions)

- 0 is the null solution \((P|0 = 0|P = P)\)
- and null molecule \((M\oplus0 = 0\oplus M = M)\) \((\tau_0; P = 0)\)

- \(X_i\) are distinct in \(E\)
- Each name \(n\) is assigned a fixed rate \(r\): \(n(r)\)

**Ex: interacting automata**

(which are CGFs using \("\mid"\) only in initial conditions):

- Automaton in state A
  \[ A = !a; A \oplus ?b; B \]
- Automaton in state B
  \[ B = !b; B \oplus ?a; A \]
- Initial conditions: 2A and 2B
  \[ A|A|B|B \]
Coding of Degradation Reactions ($A \rightarrow rA'$)

**Particles**

<table>
<thead>
<tr>
<th>[A]=1</th>
<th>[A]=2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \xrightarrow{r} \quad [A]' \quad 1$ degradation at rate $r$</td>
<td>$A \xrightarrow{r} A \xrightarrow{r} \quad [A]' \quad 2$ degradations at rate $r$</td>
</tr>
</tbody>
</table>

**Rate Law**

<table>
<thead>
<tr>
<th>[A]=1</th>
<th>[A]=2</th>
</tr>
</thead>
</table>

**Processes**

<table>
<thead>
<tr>
<th>[A]=1</th>
<th>[A]=2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A = \tau_r; A'$</td>
<td>$A = \tau_r; A'$</td>
</tr>
</tbody>
</table>

1 delay at rate $r$
(sampled from exponential distribution)

2 delays at rate $r$
(sampled from exponential distribution)

(or rather, the CGF:
$A = \tau_r A'$
$A \mid A$)
**Coding of Asymmetric Reactions ($A + B \rightarrow r A' + B'$)**

**Particles**

- $[A]=1$
- $[B]=1$

1 (productive) collision between 1+1 particles, at rate $r$

- $A \rightarrow r \rightarrow B$

**Rate Law**

$$[A]^* = -r[A][B] = -r$$

**Processes**

- $A = ?v_{(r)}; A'$
- $B = !v_{(r)}; B'$

1 interaction between 1+1 processes at rate $r$

**Particles**

- $[A]=2$
- $[B]=2$

4 (productive) collisions between 2+2 particles, at rate $r$

- $A \rightarrow r \rightarrow A$
- $B \rightarrow r \rightarrow B$

**Rate Law**

$$[A]^* = -r[A][B] = -4r$$

**Processes**

- $A = ?v_{(r)}; A'$
- $A = ?v_{(r)}; A'$
- $B = !v_{(r)}; B'$
- $B = !v_{(r)}; B'$

4 interactions between 2+2 processes at rate $r$
Coding of Symmetric Reactions ($A+A\rightarrow^r A'+A''$)

### Particles

- **[A]=1**
  - 0 collisions between 1 particle
  - $A = \ ?v \ (r/2) \ ; A'$ ⊕ $!v \ (r/2) \ ; A''$

- **[A]=2**
  - 1 collision between 2 particles at rate $r$
  - $[A]^* = -r[A](A-1) = -2r$
  - (remember: 2A are removed per collision assuming $A', A''\neq A$)

- **[A]=3**
  - 3 collisions between 3 particles at rate $r$
  - $[A]^* = -r[A](A-1) = -6r$

### Rate Law

- $[A]^* = -r[A](A-1) = \begin{cases} 0 & \text{[A]=1} \\ -2r & \text{[A]=2} \\ -6r & \text{[A]=3} \end{cases}$

### Processes

- **[A]=1**
  - 0 interactions between 1 process

- **[A]=2**
  - 2 interactions between 2 processes at rate $r/2$

- **[A]=3**
  - 6 interactions between 3 processes at rate $r/2$

### Notes

- 3 collisions at rate $r$

(assuming $A', A''\neq A$)
Symmetric Reactions: SPiM vs Matlab

\[ [x1]^* = -r[x1][(x1)-1] \quad \Leftrightarrow \quad v@r/2 \]

\[ [x2]^* = -[x2][(x2)-1]/2 \]
\[ [x1]^* = -[x1][(x1)-1] \]
\[ [x3]^* = -[x3][(x3)-1]*2 \]

Matlab

Matlab continuous system generator

SPiM

\[ A = ?v(); + !v(); \]

\[ \text{directive sample 0.01 1000} \]
\[ \text{directive plot A2(); A1(); A3()} \]
\[ \text{new v1@0.5:chan} \]
\[ \text{new v2@0.25:chan} \]
\[ \text{new v3@1.0:chan} \]
\[ \text{let A1()} = \text{do } ?v1(); \text{ or } !v1(); \]
\[ \text{let A2()} = \text{do } ?v2(); \text{ or } !v2(); \]
\[ \text{let A3()} = \text{do } ?v3(); \text{ or } !v3(); \]
\[ \text{run 1000 of (A1() | A2() | A3())} \]
Chemical Reactions to CGF

\[ \begin{align*}
  v_1: & \quad A + B \rightarrow k_1 C + C \\
  v_2: & \quad A + C \rightarrow k_2 D \\
  v_3: & \quad C \rightarrow k_3 E + F \\
  v_4: & \quad F + F \rightarrow k_4 B
\end{align*} \]

**Interaction Matrix**

<table>
<thead>
<tr>
<th></th>
<th>(v_1(k_1))</th>
<th>(v_2(k_2))</th>
<th>(v_3(k_3))</th>
<th>(v_4(k_4/2))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A)</td>
<td>(?; (C</td>
<td>C))</td>
<td>(?; D)</td>
<td></td>
</tr>
<tr>
<td>(B)</td>
<td>(!; 0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C)</td>
<td>(!; 0)</td>
<td>(?; (E</td>
<td>F))</td>
<td></td>
</tr>
<tr>
<td>(D)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(E)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(F)</td>
<td></td>
<td></td>
<td>(?; B)</td>
<td>(!; 0)</td>
</tr>
</tbody>
</table>

**Half-rate for symmetric reactions**

**Channels and rates** (1 per reaction)

**Definitions** (1 per species)

- **Degradation reaction** \(v_i: X \rightarrow_k P_i\)
  - Add \(?; P_i\) to \(<X, v_i>\).

- **Asymmetric reaction** \(v_i: X + Y \rightarrow_k P_i\)
  - Add \(?; P_i\) to \(<X, v_i>\) and \(!; 0\) to \(<Y, v_i>\).

- **Symmetric reaction** \(v_i: X + X \rightarrow_k P_i\)
  - Add \(?; P_i\) and \(!; 0\) to \(<X, v_i>\).

**Read out the resulting CGF by rows:**

- \(A = ?v_{1(k1)}(C | C) \oplus ?v_{2(k2)}D\)
- \(B = !v_{1(k1)}; 0\)
- \(C = !v_{2(k2)}; 0 \oplus \tau_{k3}(E | F)\)
- \(D = 0\)
- \(E = 0\)
- \(F = ?v_{4(k4/2)}B \oplus !v_{4(k4/2)}; 0\)
That \textit{CGF in SPiM}\\

\[\begin{align*}
A &= ?v_{1(k1)};(C|C) \oplus ?v_{2(k2)};D \\
B &= !v_{1(k1)};0 \\
C &= !v_{2(k2)};0 \oplus \tau_{k3};(E|F) \\
D &= 0 \\
E &= 0 \\
F &= ?v_{4(k4/2)};B \oplus !v_{4(k4/2)};0
\end{align*}\]

directive sample 10.0

directive plot \texttt{A(); B(); C(); D(); E(); F()}

\texttt{val k1 = 0.001  new v1@k1:chan}
\texttt{val k2 = 0.001  new v2@k1:chan}
\texttt{val k3 = 1.0}
\texttt{val k4 = 0.001  new v4@k4/2.0:chan}

\texttt{let A() = do ?v1;(C(C)) or ?v2;D()}
\texttt{and B() = !v1}
\texttt{and C() = do !v2 or delay@k3;(E|F())}
\texttt{and D() = ()}
\texttt{and E() = ()}
\texttt{and F() = do ?v4;B() or !v4}

\texttt{run 300 of (A|B|C|D|E|F())}
... in other words

From chemical reactions $C$ to a CGF $\Pi(C)$:

$$\Pi(C) = \{(X = \bigoplus((\nu_i: X \rightarrow^k P) \in C) \circ \tau_k; P) \oplus$$
$$\bigoplus((\nu_i: X+Y \rightarrow^k P) \in C \text{ and } Y \neq X) \circ (\nu_{i(k)}; 0) \oplus$$
$$\bigoplus((\nu_i: Y+X \rightarrow^k P) \in C \text{ and } Y \neq X) \circ (\nu_{i(k)}; 0) \oplus$$
$$\bigoplus((\nu_i: X+X \rightarrow^k P) \in C) \circ (\nu_{i(k/2)}; P \oplus \nu_{i(k/2)}; 0) \}$$

s.t. $X$ is a species in $C$}
From Processes
Directly to ODEs (hard)
“Micromodels”: Continuous Time Markov Chains

- The underlying semantics of stochastic π-calculus (and stochastic interacting automata). Well established in many ways.
  - Automata with rates on transitions.

- “The” correct semantics for chemistry, executable.
  - Gillespie stochastic simulation algorithm

- Good description of “individual” behavior

- But does not give a good sense of “collective” properties.
  - Yes one can do simulation.
  - Yes one can do program analysis.
  - Yes one can do model checking.
  - But somewhat lacking in “predictive power”.
Micromodels have lots of advantages
- Compositional, compact, mechanistic, etc.

But they always ask:
- “Yes, but how does your automata model relate to the 75 ODE models in the literature?”

From processes/automata to ODEs directly:
- In principle: just write down the Rate Equation:
  - Determine the set of all possible states $S$ of each process.
  - Determine the rates of the transitions between such states.
  - Let $[S]$ be the “number of processes in state $S$” as a function of time.
  - Define for each state $S$:
    $$[S]^* = \text{(rate of change of the number of processes in state $S$)}$$
    Cumulative rate of transitions from any state $S'$ to state $S$, times $[S']$,
    minus cumulative rate of transitions from $S$ to any state $S''$, times $[S]$.
  - Intuitive (rate = inflow minus outflow), but often clumsy to write down precisely.

But why go directly from processes to ODE?
- If we first convert processes to chemical reactions, then we can convert to ODEs by standard means!
From Processes to Chemistry
Automata to Chemistry (by hand)
Automata to Chemistry

\[
\begin{align*}
A & \rightarrow B + B \\
B + A & \rightarrow A + A
\end{align*}
\]

\[
\begin{align*}
A + B_d & \rightarrow B + B_d \\
B + A_d & \rightarrow A + A_d
\end{align*}
\]

\[
\begin{align*}
A + C & \rightarrow C + C \\
C + B & \rightarrow B + B \\
B + A & \rightarrow A + A
\end{align*}
\]

\[
\begin{align*}
A + C_d & \rightarrow C + C_d \\
C + B_d & \rightarrow B + B_d \\
B + A_d & \rightarrow A + A_d
\end{align*}
\]
(Lack of) Compositionality

6 states

6 reactions

+2 states

+6 reactions ??

6 states

6 reactions

+3 states

+6 reactions ??

A+B → A₁+B
A₁+B → A₂+B
A₂+B → B+B
B+A → B₁+A
B₁+A → B₂+A
B₂+A → A+A

A+C → A₁+C
A₁+C → C+C
C+B → C₁+B
C₁+B → B+B
B+A → B₁+A
B₁+A → A+A

Doping

Doping

Doping

A+B_d → A₁+B_d
A₁+B_d → A₂+B_d
A₂+B_d → B+B_d
B+A_d → B₁+A_d
B₁+A_d → B₂+A_d
B₂+A_d → A+A_d
Single Automata Models are $n^2$ More Compact

### Automaton

- $S_0 \rightarrow \text{!a}$
- $S_1 \rightarrow \text{!a}$
- $S_2 \rightarrow \text{!a}$
- $S_3 \rightarrow \text{!a}$

### Chemistry

- $S_0 + S_0 \rightarrow S_0 + S_1$
- $S_0 + S_1 \rightarrow S_0 + S_2$
- $S_0 + S_2 \rightarrow S_0 + S_0$
- $S_1 + S_0 \rightarrow S_1 + S_1$
- $S_1 + S_1 \rightarrow S_1 + S_2$
- $S_1 + S_2 \rightarrow S_1 + S_0$
- $S_2 + S_0 \rightarrow S_2 + S_1$
- $S_2 + S_1 \rightarrow S_2 + S_2$
- $S_2 + S_2 \rightarrow S_2 + S_0$

### Comparison

- 3 states
  - (2*3 transitions)
  - $3^2$ reactions

- 4 states
  - (2*4 transitions)
  - $4^2$ reactions
Process Normal Forms to Chemistry
Chemical Parametric Form (CPF)

E ::= X_1(p_1)=M_1, ..., X_n(p_n)=M_n

M ::= π_1;P_1 ⊕ ... ⊕ π_n;P_n

P ::= X_1(p_1) | ... | X_n(p_n)

π ::= τ_r ?n(p) !n(p)

CPF ::= E,P

Definitions (n ≥ 0)

Molecules (n ≥ 0)

Solutions (n ≥ 0)

Interactions

with initial conditions

⊕ is stochastic choice (vs. + for chemical reactions)
0 is the null solution (P|0 = 0|P = P)
and null molecule (M⊕0 = 0⊕M = M) (τ_0;P = 0)
X_i are distinct in E, p are vectors of names
p are vectors of distinct names when in binding position
Each free name n in E is assigned a fixed rate r:
written either n_(r), or ρ_{CPF}(n)=r.

CPF is NF without name generation.

CPF is CGF plus parameters.

Random example

P(a,b) = ?a(c); (P(c,b) | Q(c))
Q(a) = !a(a); 0
P(x_(r),y_(s)) | Q(x_(r))

Repressilator

Neg(a,b) = ?a; Inh(a,b) ⊕ τ_ε; (Tr(b) | Neg(a,b))
Inh(a,b) = τ_η; Neg(a,b)
Tr(b) = !b; Tr(b) ⊕ τ_δ; 0
Neg(x,y) | Neg(y,z) | Neg(z,x)
CPF to Chemistry: the CGF Case

CGF = The CPF subset with no communication, no parametric definitions.

\[
\begin{align*}
E & := X_1=M_1, \ldots, X_n=M_n \\
M & := \pi_1;P_1 \oplus \ldots \oplus \pi_n;P_n \\
P & := X_1 | \ldots | X_n \\
\pi & := \tau_r \ ?n_r \ !n_r \\
CGF & := E, P
\end{align*}
\]

Definitions (n≥0)

Molecules (n≥0)

Solutions (n≥0)

Interactions (delay, input, output)

Definitions with Initial Conditions

Each X in E is seen as a separate species.

Chemical reactions for E:

(N.B.: \{\ldots\}^m is a multiset, and P is P with all the | changed to +)

\[
\begin{align*}
\text{Ch}_G(E) & := \{(X \rightarrow^r P) \text{ s.t. } (X \equiv \tau_r;P \oplus \ldots) \in E\}^m \\
& \cup^m \{(X + Y \rightarrow^r P + Q) \text{ s.t. } X \neq Y, (X \equiv \ ?n_r(P \oplus \ldots), (Y \equiv !n_r(Q \oplus \ldots)) \in E^2\}^m \\
& \cup^m \{(X + X \rightarrow^{2r} P + Q) \text{ s.t. } (X \equiv \ ?n_r;P \oplus \ldots \equiv !n_r;Q \oplus \ldots) \in E\}^m
\end{align*}
\]

Initial conditions for P:

\[
\text{Ch}_G(P) := P
\]
(Note on computing the multisets)

A multiset \( M \in \text{Multiset}(S) \), where \( S \) is a set with equality, is a total function \( S \rightarrow \text{Nat} \), which may also be written as a finite enumeration with repetitions: \{...\}^m.

Multiset binary union is the function \( \bigcup^m(M,M') = \{s \mid M(s)+M'(s)\} \).
Multiset big union over a finite index set \( S \) is written: \( \bigcup^m _i \in S \) of \( M_i \).

- The shorthand: \{\( X \rightarrow r \) P s.t. \( X \equiv \tau_r;P \oplus ... \) \} \in E}^m
means: "for each \( (X=\pi_1;P_1\oplus...\oplus\pi_n;P_n) \in E \) and for each \( i \) such that \( \pi_i=\tau_r \), return a copy of \( (X \rightarrow r P_i) \)."

i.e., it is defined as the following finite union of singleton multisets:
\[ \bigcup^m \{ (X=\pi_1;P_1\oplus...\oplus\pi_n;P_n) \in E \} \text{ of } \{ (X \rightarrow r P_i) \}^m \]

- The shorthand: \{(X + Y \rightarrow r P + Q) s.t. X\#Y, (X \equiv ?n(r);P \oplus ...), (Y \equiv !n(r);Q \oplus ...) \in E^2}^m
means: "for each ordered pair \( (X=\pi_1;P_1\oplus...\oplus\pi_n;P_n),(Y=\sigma_1;Q_1\oplus...\oplus\sigma_n;Q_n) \) from \( E^2 \) with \( X\#Y \), for each \( n(r) \), for each \( i \) such that \( \pi_i=?n(r) \), and for each \( j \) such that \( \sigma_j=!n(r) \), return a copy of the reaction \((X + Y \rightarrow r P_i + Q_j)\)"

i.e., it is defined as the following finite union of singleton multisets:
\[ \bigcup^m \langle (X=\pi_1;P_1\oplus...\oplus\pi_n;P_n),(Y=\rho_1;Q_1\oplus...\oplus\rho_m;Q_m) \rangle \in E^2 \text{ with } X\#Y \\
\text{of } \{ (X + Y \rightarrow r P_i + Q_j) \}^m \]

- The shorthand: \{(X + X \rightarrow 2r P + Q) s.t. (X \equiv ?n(r);P \oplus ... \equiv !n(r);Q \oplus ...) \in E}^m
means: "for each \( (X=\pi_1;P_1\oplus...\oplus\pi_n;P_n) \text{ in } E \), for each \( n(r) \), for each \( i \) such that \( \pi_i=?n(r) \), and for each \( j \) such that \( \pi_j=!n(r) \), return a copy of the reaction \((X + X \rightarrow 2r P_i + P_j)\)"
**Example**

\[
\begin{align*}
\text{E,P:} & & \text{Ch}_G(E,P): \\
\text{Na} &= ?i; \text{Na}^+ & \text{Na} + \text{Cl} & \rightarrow_{\rho(i)} \text{Na}^+ + \text{Cl}^- \\
\text{Cl} &= !i; \text{Cl}^- & \text{Na}^+ + \text{Cl}^- & \rightarrow_{\rho(d)} \text{Na} + \text{Cl} \\
\text{Na}^+ &= ?d; \text{Na} & \text{Na} + \text{Na} + \text{Cl} + \text{Cl} & \rho_E(n_r) = r \\
\text{Cl}^- &= !d; \text{Cl} & & \end{align*}
\]

\[
\text{Ch}_G(E) := \ldots \\
\cup_m \{(X + Y \rightarrow r P + Q) \text{ s.t. } X \neq Y, \langle (X \equiv \mathcal{I} n_r; P \oplus \ldots), (Y \equiv \mathcal{O} n_r; Q \oplus \ldots) \rangle \in E^2 \}^m \\
\cup_m \ldots
\]
Subtler Examples

These are not finite state systems, but finite species systems are ok!

\[ \text{E: } \quad X = \tau_r(X | X) \quad \text{C(E): } \quad X \rightarrow_r X + X \quad \text{Unbounded state, but only 1 species. No problem!} \]

**Multisets:**
The same interaction can occur multiple times and must be taken into account:

\[ \text{E: } \quad A = ?n;B \oplus ?n;B \quad C = !n;D \quad \text{C(E): } \quad A + C \rightarrow^{\rho(n)} B + D \quad A + C \rightarrow^{\rho(n)} B + D \quad \text{That is: } \quad A + C \rightarrow^{2\rho(n)} B + D \]

**Symmetric reactions:**

\[ \text{E: } \quad X = !a;0 \oplus ?a;Y \quad \text{C(E): } \quad X + X \rightarrow^{2\rho(a)} Y \]

The rate of a was pre-halved and must be restored.
CPF to Chemistry: Handling Parameters

Consider first the CPF subset with no communication (pure $?n, !n$).

**Grounding** (replace parameters with constants)

where $X/p$ is a name in bijection with $<X,p>$
(each $X/p$ is seen as a separate species)

\[
\frac{\pi_1;P_1 \oplus \ldots \oplus \pi_n;P_n}{(\pi_1;P_1) \oplus \ldots \oplus (\pi_n;P_n)} \overset{\text{def}}{=} \pi_1;(\frac{P_1}{}) \oplus \ldots \oplus \pi_n;(\frac{P_n}{})
\]

\[
\frac{X_1(p_1) \mid \ldots \mid X_n(p_n)}{(X_1(p_1) \mid \ldots \mid X_n(p_n))} \overset{\text{def}}{=} \frac{X_1/p_1}{\ldots \mid \frac{X_n/p_n}{}}
\]

Let $N$ be the set of free names occurring in $E$ (names not bound by definitions).

$E_G$ is the **Parametric Explosion** of $E$ (still a finite species system)
computed by replacing parameters with all combinations of free names in $E$

\[
E_G := \{(X/q = (M[p\leftarrow q])) \text{ s.t. } (X(p) = M) \in E \text{ and } q \in N^\#p\}
\]

$P_G := \frac{}{P}$  \hspace{1cm} (simply ground the given initial conditions once)

$E_G$ is a CGF! To obtain the chemical reactions $Ch_p(E)$, just compute $Ch_G(E_G)$

$Ch_p(E) = Ch_G(E_G)$
Example

$E,P$ (with free names $x,y$):

\[
\begin{align*}
P(a,b) &= \text{?}a; (P(a,b) \mid Q(a)) \\
Q(a) &= \text{!}a; 0 \\
P(x,y) \mid Q(x)
\end{align*}
\]

$E_G,P_G$:

\[
\begin{align*}
Q/x &= \text{!}x; 0 \\
Q/y &= \text{!}y; 0 \\
P/x,y &= \text{?}x; (P/x,y \mid Q/x) \\
P/y,x &= \text{?}y; (P/y,x \mid Q/y) \\
P/x,x &= \text{?}x; (P/x,x \mid Q/x) \\
P/y,y &= \text{?}y; (P/y,y \mid Q/y) \\
P/x,y \mid Q/x
\end{align*}
\]

Parametric Explosion

$C$:

\[
\begin{align*}
Q/x + P/x,y &\rightarrow^{\rho(x)} P/x,y + Q/x \\
Q/x + P/x,x &\rightarrow^{\rho(x)} P/x,x + Q/x \\
Q/y + P/y,x &\rightarrow^{\rho(y)} P/y,x + Q/y \\
Q/y + P/y,y &\rightarrow^{\rho(y)} P/y,y + Q/y \\
P/x,y + Q/x
\end{align*}
\]

Chemical Reactions

Reduced $C$ (from initial conditions)

\[
\begin{align*}
Q/x + P/x,y &\rightarrow^{\rho(x)} P/x,y + Q/x \\
P/x,y + Q/x
\end{align*}
\]

Note that $E_G$ is not necessarily the “full” explosion (there may not be enough free variables in $E,P$ to generate it). In particular, if $E,P$ is closed (no free variables), then $E_G$ contains only the parameterless definitions, which must then be closed and hence unrelated to any parametric definitions. One can add initial conditions $X_1(p_1) \mid \ldots \mid X_n(p_n)$ where $X_i$ are all the definitions in $E$ and $p_1 \ldots p_n$ are made of distinct (free) variables. Then $E_G$ becomes the most general set of chemical reactions from $E$, out of any possible initial conditions.
Iterative Algorithm

The chemical reactions for $E$, computed directly from the parametric explosion $E_G$ are highly redundant because they include all the parameter permutation symmetries. The following iterative algorithm, for the parametric case, computes a subset of $E_G$ from the initial conditions of $E_P$. It produces a (usually) much smaller although not necessarily minimal set $C$.

Here:
- $C$ is the incrementally built set of chemical reactions
- $E_c$ (a CGF!) is the incrementally built set of grounded definitions from the original $E$.

**Initialization**

$E_C := \{(X/q = /(M\{p\leftarrow q\})) \text{ s.t. } X(q) \text{ occurs in } P \text{ and } (X(p) = M) \in E\}$ (from initial conditions $P$)

**Iteration**

$C := Ch_G(E_C)$

$E_C' := E_C \cup \{X/q = /(M\{p\leftarrow q\})) \text{ s.t. } X/q \text{ occurs in } C \text{ and } (X(p) = M) \in E\}$

**Termination**

if $E_C' = E_C$ then stop and return $(C,/(P))$ else $E_C := E_C'$ and iterate.

The algorithm terminates because $E_C$ never shrinks and is always a subset of $E_G$, which is finite.
Example: Neg(x,x)

\[ E = \]
\[
\text{Neg}(a,b) = ?a; \text{Inh}(a,b) \oplus \tau \varepsilon; (\text{Tr}(b) \mid \text{Neg}(a,b)) \\
\text{Inh}(a,b) = \tau \eta; \text{Neg}(a,b) \\
\text{Tr}(b) = !b; \text{Tr}(b) \oplus \tau \delta; 0 \\
\text{Neg}(x,x)
\]

----- initialization ----- 
\[ E_c := \{ \text{Neg/x,x} = ?x; \text{Inh/x,x} \oplus \tau \varepsilon; (\text{Tr/x} \mid \text{Neg/x,x}) \} \]

----- iteration 1 ----- 
\[ C := \{ \text{Neg/x,x} \rightarrow^e \text{Tr/x} + \text{Neg/x,x} \} \]
\[ E_c := \{ \text{Neg/x,x} = ?x; \text{Inh/x,x} \oplus \tau \varepsilon; (\text{Tr/x} \mid \text{Neg/x,x}) \\
\text{Tr/x} = !x; \text{Tr/x} \oplus \tau \delta; 0 \} \]

----- iteration 2 ----- 
\[ C := \{ \text{Neg/x,x} \rightarrow^e \text{Tr/x} + \text{Neg/x,x} \}
\text{Tr/x} \rightarrow^\delta 0 \\
\text{Tr/x} + \text{Neg/x,x} \rightarrow^{\rho(x)} \text{Tr/x} + \text{Inh/x,x} \\
\text{Inh/x,x} \rightarrow^\eta \text{Neg/x,x} \]
\[ E_c := \{ \text{Neg/x,x} = ?x; \text{Inh/x,x} \oplus \tau \varepsilon; (\text{Tr/x} \mid \text{Neg/x,x}) \\
\text{Tr/x} = !x; \text{Tr/x} \oplus \tau \delta; 0 \\
\text{Inh/x,x} = \tau \eta; \text{Neg/x,x} \} \]

----- iteration 3 ----- 
\[ C := \{ \text{Neg/x,x} \rightarrow^e \text{Tr/x} + \text{Neg/x,x} \}
\text{Tr/x} \rightarrow^\delta 0 \\
\text{Tr/x} + \text{Neg/x,x} \rightarrow^{\rho(x)} \text{Tr/x} + \text{Inh/x,x} \]
\[ E_c := \text{no change} \]

----- termination ----- 
\[ \text{Neg/x,x} \rightarrow^e \text{Tr/x} + \text{Neg/x,x} \]
\[ \text{Tr/x} \rightarrow^\delta 0 \]
\[ \text{Tr/x} + \text{Neg/x,x} \rightarrow^{\rho(x)} \text{Tr/x} + \text{Inh/x,x} \]
\[ \text{Inh/x,x} \rightarrow^\eta \text{Neg/x,x} \]
\[ \text{Neg/x,x} \]
Example: $\text{Tr}(x) \mid \text{Neg}(x, y)$

\[
\begin{align*}
E = & \quad \text{Neg}(a, b) = ?a; \text{Inh}(a, b) \oplus \tau_e; (\text{Tr}(b) \mid \text{Neg}(a, b)) \\
& \quad \text{Inh}(a, b) = \tau_\eta; \text{Neg}(a, b) \\
& \quad \text{Tr}(b) = !b; \text{Tr}(b) \oplus \tau_\delta; 0 \\
& \quad \text{Tr}(x) \mid \text{Neg}(x, y)
\end{align*}
\]

----- initialization ----- 
\[
E^c := \{ \text{Tr}/x = !x; \text{Tr}/x \oplus \tau_\delta; 0 \\
\quad \text{Neg}/x,y = ?x; \text{Inh}/x,y \oplus \tau_e; (\text{Tr}/y \mid \text{Neg}/x,y) \}
\]

----- iteration 1 ----- 
\[
C := \{ \text{Tr}/x \rightarrow_\delta 0 \\
\quad \text{Neg}/x,y \rightarrow^e \text{Tr}/y + \text{Neg}/x,y \\
\quad \text{Tr}/x + \text{Neg}/x,y \rightarrow^{\rho(x)} \text{Tr}/x + \text{Inh}/x,y \}
\]
\[
E^c := \{ \text{Tr}/x = !x; \text{Tr}/x \oplus \tau_\delta; 0 \\
\quad \text{Neg}/x,y = ?x; \text{Inh}/x,y \oplus \tau_e; (\text{Tr}/y \mid \text{Neg}/x,y) \\
\quad \text{Tr}/y = !y; \text{Tr}/y \oplus \tau_\delta; 0 \\
\quad \text{Inh}/x,y = \tau_\eta; \text{Neg}/x,y \}
\]

----- iteration 2 ----- 
\[
C := \{ \text{Tr}/x \rightarrow_\delta 0 \\
\quad \text{Neg}/x,y \rightarrow^e \text{Tr}/y + \text{Neg}/x,y \\
\quad \text{Tr}/x + \text{Neg}/x,y \rightarrow^{\rho(x)} \text{Tr}/x + \text{Inh}/x,y \\
\quad \text{Tr}/y \rightarrow_\delta 0 \\
\quad \text{Inh}/x,y \rightarrow^\eta \text{Neg}/x,y \}
\]
\[
E^c := \text{no change}
\]

----- termination ----- 
\[
\begin{align*}
\text{Tr}/x & \rightarrow_\delta 0 \\
\text{Neg}/x,y & \rightarrow^e \text{Tr}/y + \text{Neg}/x,y \\
\text{Tr}/x + \text{Neg}/x,y & \rightarrow^{\rho(x)} \text{Tr}/x + \text{Inh}/x,y \\
\text{Tr}/y & \rightarrow_\delta 0 \\
\text{Inh}/x,y & \rightarrow^\eta \text{Neg}/x,y \\
\text{Tr}/x + \text{Neg}/x,y
\end{align*}
\]
CPF to Chemistry: Handling Communication

**Grounding** (replace parameters with constants)
just one main change: now also convert each input parameter into a ground choice of all possible inputs

- \( N \) is the set of free names in \( E, P \)
- \( \#p \) is the length of \( p \)
- \( n/p \) is a name in bijection with \( <n, p> \)
- \( X/p \) is a name in bijection with \( <X, p> \)
  (each \( X/p \) is seen as a separate species)

\[
/_{N}(\tau_r; P) = \tau_r; /_{N}(P)
\]
\[
/_{N}(!n_{(r)}(p); P) = !n/p_{(r)}; /_{N}(P)
\]
\[
/_{N}(?n_{(r)}(p); P) = \oplus(q \in N^{\#p}) \text{ of } ?n/q_{(r)}; /_{N}(P[p \leftarrow q])
\]
\[
/_{N}(\pi_1; P_1 \oplus ... \oplus \pi_n; P_n) = /_{N}(\pi_1; P_1) \oplus ... \oplus /_{N}(\pi_n; P_n)
\]
\[
/_{N}(X_1(p_1) | ... | X_n(p_n)) = X_1/p_1 | ... | X_n/p_n
\]

\( E_G \) is again the **Parametric Explosion** of \( E \)

\[
E_G := \{(X/q = /_{N}(M[p \leftarrow q])) \text{ s.t. } (X(p) = M) \in E \text{ and } q \in N^{\#p}\}
\]
\[
P_G := /_{N}(P)
\]  
(simply ground the given initial conditions once)

\( \text{Ch}(E) = \text{Ch}_G(E_G) \) \( E_G \) is a again a CGF!
Example

E (with free names x,y):
- \( P(a,b) = ?a(c); (P(c,b) \parallel Q(c)) \)
- \( Q(a) = !a(a); 0 \)
- \( P(x,y) \parallel Q(x) \)

\[ \text{Parametric Explosion} \]

\[ \text{Chemical Reactions} \]

\[ \text{Reduced } C \text{ (from initial conditions)} \]

E\(_G\):
- \( Q/x = !x/x; 0 \)
- \( Q/y = !y/y; 0 \)
- \( P/x,y = ?x/x; (P/x,y \parallel Q/x) \oplus ?x/y; (P/y,y \parallel Q/y) \)
- \( P/y,x = ?y/x; (P/x,x \parallel Q/x) \oplus ?y/y; (P/y,y \parallel Q/y) \)
- \( P/x,x = ?x/x; (P/x,x \parallel Q/x) \oplus ?x/y; (P/y,y \parallel Q/y) \)
- \( P/y,y = ?y/x; (P/x,y \parallel Q/x) \oplus ?y/y; (P/y,y \parallel Q/y) \)
- \( P/x,y \parallel Q/x \)

C:
- \( Q/x + P/x,y \rightarrow_{\rho(x)} P/x,y + Q/x \)
- \( Q/x + P/x,x \rightarrow_{\rho(x)} P/x,x + Q/x \)
- \( Q/y + P/y,x \rightarrow_{\rho(y)} P/y,x + Q/y \)
- \( Q/y + P/y,y \rightarrow_{\rho(y)} P/y,y + Q/y \)
- \( P/x,y + Q/x \)

\[ \text{Parametric Explosion} \]

\[ \text{Chemical Reactions} \]

\[ \text{Reduced } C \text{ (from initial conditions)} \]
Example

E (with free names x,y):

\[
P(a,b) = ?a(c); (P(c,b) \mid Q(c,c))
\]

\[
Q(a,b) = !a(a); 0 \oplus !b(b); 0
\]

\[
P(x,y) \mid Q(x,y)
\]

\[E_G:
\]

\[
Q/x,y = !x/x; 0 \oplus !y/y; 0
\]

\[
Q/y,x = !y/y; 0 \oplus !x/x; 0
\]

\[
Q/x,x = !x/x; 0 \oplus !x/x; 0
\]

\[
Q/y,y = !y/y; 0 \oplus !y/y; 0
\]

\[
P/x,y = ?x/x; (P/x,y \mid Q/x,x) \oplus ?x/y; (P/y,y \mid Q/y,y)
\]

\[
P/y,x = ?y/x; (P/x,x \mid Q/x,x) \oplus ?y/y; (P/y,x \mid Q/y,y)
\]

\[
P/x,x = ?x/x; (P/x,x \mid Q/x,x) \oplus ?x/y; (P/y,x \mid Q/y,y)
\]

\[
P/y,y = ?y/x; (P/x,y \mid Q/x,x) \oplus ?y/y; (P/y,y \mid Q/y,y)
\]

\[
P/x,y \mid Q/x
\]

\[C:
\]

\[
Q/x,y + P/x,y \rightarrow^\rho(x) P/x,y + Q/x,x
\]

\[
Q/x,y + P/x,x \rightarrow^\rho(x) P/x,x + Q/x,x
\]

\[
Q/x,y + P/y,x \rightarrow^\rho(y) P/y,x + Q/y,y
\]

\[
Q/x,y + P/y,y \rightarrow^\rho(y) P/y,y + Q/y,y
\]

\[
Q/y,x + P/y,x \rightarrow^\rho(y) P/y,x + Q/y,y
\]

\[
Q/y,x + P/y,y \rightarrow^\rho(y) P/y,y + Q/y,y
\]

\[
Q/y,x + P/x,x \rightarrow^\rho(x) P/x,x + Q/x,x
\]

\[
Q/y,x + P/x,y \rightarrow^\rho(x) P/x,y + Q/x,x
\]

\[
Q/y,y + P/x,y \rightarrow^\rho(x) P/x,y + Q/x,x
\]

\[
Q/y,y + P/y,x \rightarrow^\rho(y) P/y,x + Q/y,y
\]

\[
Q/y,y + P/y,y \rightarrow^\rho(y) P/y,y + Q/y,y
\]

\[
P/x,y + Q/x,y
\]

reduced C (from initial conditions)

\[
Q/x,y + P/x,y \rightarrow^\rho(x) P/x,y + Q/x,x
\]

\[
Q/x,x + P/x,y \rightarrow^2\rho(x) P/x,y + Q/x,x
\]

\[
P/x,y + Q/x,y
\]
Exponential Explosion via Communication

Parametric form (size 3).

\[ X(t,f) = !t(f);X(f,t) \]
\[ A(x_1,x_2,x_3) = \]
\[ ?x_1(y);A(y,x_2,x_3) \oplus \]
\[ ?x_2(y);A(x_1,y,x_3) \oplus \]
\[ ?x_3(y);A(x_1,x_2,y) \]
\[ A(t_1,t_2,t_3) | \]
\[ X(t_1,f_1) | X(t_2,f_2) | X(t_3,f_3) \]

green = free names with associated rates

Exponentially larger ground form (size 8).

\[ X/t_1,f_1 = !t_1/f_1; X/f_1,t_1 \]
\[ X/f_1,t_1 = !f_1/t_1; X/t_1,f_1 \]
\[ X/t_2,f_2 = !t_2/f_2; X/f_2,t_2 \]
\[ X/f_2,t_2 = !f_2/t_2; X/t_2,f_2 \]
\[ X/t_3,f_3 = !t_3/f_3; X/f_3,t_3 \]
\[ X/f_3,t_3 = !f_3/t_3; X/t_3,f_3 \]
\[ A/t_1,t_2,t_3 = \]
\[ ?t_1/f_1;A/f_1,t_1,t_2,t_3 \oplus \]
\[ ?t_2/f_2;A/t_1,f_2,t_3 \oplus \]
\[ ?t_3/f_3;A/t_1,t_2,f_3 \]
\[ A/f_1,t_2,t_3 = \]
\[ ?f_1/t_1;A/t_1,t_2,t_3 \oplus \]
\[ ?t_2/f_2;A/t_1,f_2,t_3 \oplus \]
\[ ?t_3/f_3;A/t_1,t_2,f_3 \]
\[ A/t_1,t_2,t_3 | \]
\[ X/t_1,f_1 | X/t_2,f_2 | X/t_3,f_3 \]

We have not fully expanded the input summations in \( A(\ldots) \): only enough for the channels that are actually used: e.g., there would never be a communication on channel \( t_1/f_2 \).
CPF Iterative Algorithm

The same as before, just using the new grounding $\mathcal{N}$
(based on the set $\mathcal{N}$ of free names of $E,P$, for expanding inputs).

initialization

\[ N = \text{fn}(E,P) \]
\[ E_C := \{ (X/p = $\mathcal{N}(M{p\leftarrow q})) \text{ s.t. } X(q) \text{ occurs in } P \text{ and } (X(p) = M) \in E \} \quad \text{(initial conditions)} \]

iteration

\[ C := \text{Ch}_E(E_C) \]
\[ E_C' := E_C \cup \{ (X/p = $\mathcal{N}(M{p\leftarrow q})) \text{ s.t. } X/q \text{ occurs in } C \text{ and } (X(p) = M) \in E \} \]

termination

if $E_C' = E_C$ then stop and return $(C,\mathcal{N}(P))$ else $E_C := E_C'$ and iterate.
A General $\pi$-calculus Normal Form

\[ E ::= X_1(p_1)=\nu q_1 M_1, \ldots, X_n(p_n)=\nu q_n M_n \]
\[ M ::= \pi_1; P_1 \oplus \ldots \oplus \pi_n; P_n \]
\[ P ::= X_1(p_1) | \ldots | X_n(p_n) \]
\[ \pi ::= \tau_r \ ? n(p) \ ! n(p) \]
\[ NF ::= E, P \]

Definitions \((n \geq 0)\)

Molecules \((n \geq 0)\)

Solutions \((n \geq 0)\)

Interactions

with initial conditions

Any $\pi$-calculus process can be written in this normal form.
But not all of those can be translated to chemical reactions, because we need to restrict to a finite number of species.

Example (complexation):
\[
\begin{align*}
P(a) &= (\nu n(d)) ! a(n); P'(a,n) \\
P'(a,n) &= ! n; P(a) \\
Q(a) &= ? a(n); Q'(a,n) \\
Q'(a,n) &= ? n; Q(a) \\
P(x_{(c)}) &| Q(x_{(c)})
\end{align*}
\]

Example (enzymatic reaction):
\[
\begin{align*}
E(a) &= (\nu n(d), m(p)) ! a(n,m); E'(a,n,m) \\
E'(a,n,m) &= ! n; E(a) \oplus ! m; E(a) \\
S(a) &= ? a(n,m); S'(a,n,m) \\
S'(a,n,m) &= ? n; S(a) \oplus ? m; P() \\
P() &= \tau_0; 0 \\
E(x_{(c)}) &| S(x_{(c)}) | S(x_{(c)})
\end{align*}
\]
Summary

- **From Chemistry to ODEs**
  - Via the stoichiometric matrix

- **From Chemistry to Processes**
  - Via the interaction matrix

- **From Processes to Chemistry**
  - Ground processes: by analysis of normal form interactions
  - Parametric processes: by parametric explosion

- **Compositionality**
  - Processes are compositional in components (you just “add components”) and chemistry is compositional in reactions (you just “add reactions”). But:
    - Adding reactions has a *linear* effect on the number of components (adding one reaction adds one or two more process actions).
    - Adding components has a *quadratic* effect on the number of reactions (adding one species may add one reaction for each species).
    - ODEs grow quadratically along with the chemical reactions (stoichiometric matrix has one column per reaction).
  - Parameterization is available only on processes.
Q?