1 ideas
idea I

Two aspects in solving a distributed problem:
- local steps towards a solution
- backtracking (deadlock escape)

Sequential case: can try to always make progress to solution, but NP! // case: one has to!
idea I - continued

backtrack -> infrastructure (make it a “library”)  
code easier to prove and understand

universal backtrack strategy

\[ p \rightarrow \Gamma \cdot p \]

i.e., add history to a process
results/reversible CCS

1. Universal cover ppty:
   distributed history characterizes traces up to concurrency

2. Weak-bisimulation:
   rev\(\langle p \rangle\) + irreversible actions / causal transition
   system\(\langle p \rangle\) - only irreversible actions observable

3. Syntax-independent history construction (eg works for Petri nets, pi-calculus)
Could relax universal cover ppty:

introduce flex-moves (never a choice) not memorized weak memories: forget synch partner

... not done anything!
idea II

∞-hesitation, efficiency \nProbabilization of \text{rev}(p)

Exhaustivity \nProbabilistic equilibrium
idea III

What prob structure? borrow from stat phys

distributed CT Metropolis

build a potential energy function
drive kinetics (Newtonian style, stochastic version)

build a causal/concurrent, and convergent potential energy on the state space of reversible CCS
the reversible CCS transition system
Reversible Communicating Processes

\[ \Gamma \cdot (p_1, \ldots, p_n) \rightarrow_f \Gamma_1 \cdot p_1, \ldots, \Gamma_n \cdot p_n \]

\text{n-fork}

\[ \Gamma_1 \cdot (a_1 p_1 + q_1), \ldots, \Gamma_m \cdot (a_m p_m + q_m) \rightarrow_s^a \]

\[ \Gamma_1(\vec{\Gamma}, a_1, q_1) \cdot p_1, \ldots, \Gamma_m(\vec{\Gamma}, a_m, q_m) \cdot p_m \]

With a unique naming scheme and enough info to reverse uniquely
Symmetric TS (so strongly connected)

“Simplicity” of TS: at most one jump

Slight pb with sums between x and y

Near acyclic

Countable state space (recursion)
potential/rate ratio constraint

\[ \frac{q(y, x)}{q(x, y)} = \frac{p(y)}{p(x)} = e^{-(V(y)-V(x))} \]

\[ \sum_x e^{-V(x)} < \infty \]

that is convergence by def!

NB: lower energy/higher probability
**EXPLOSIVE GROWTHS**

$q \rightarrow^f 0 \cdot p(a), 1 \cdot p(\bar{a})$

$\rightarrow^{fs} 0a0 \cdot p(a), 0a1 \cdot p(a), 1\bar{a}0 \cdot p(\bar{a}), 1\bar{a}1 \cdot p(\bar{a})$

$= 0a0 \cdot a(p(a), p(a)), 0a1 \cdot a(p(a), p(a)), 1\bar{a}0 \cdot \bar{a}(p(\bar{a}), p(\bar{a})), 1\bar{a}1 \cdot \bar{a}(p(\bar{a}), p(\bar{a}))$

$\rightarrow^{fs} 0a0a0 \cdot p(a), 0a0a1 \cdot p(a), 0a1a0 \cdot p(a), 0a1a1 \cdot p(a), 4, 4$  

$1\bar{a}0\bar{a}0 \cdot p(\bar{a}), 1\bar{a}0\bar{a}1 \cdot p(\bar{a}), 1\bar{a}1\bar{a}0 \cdot p(\bar{a}), 1\bar{a}1\bar{a}1 \cdot p(\bar{a})$

$\ldots$

$\rightarrow^{fs} \prod_{w \in 2^k} 0w(a) \cdot p(a), \prod_{w \in 2^k} 1w(\bar{a}) \cdot p(\bar{a})$

<table>
<thead>
<tr>
<th>Event Horizon</th>
<th>Nb of Complete Matchings</th>
<th>Upper Bound on the Number of Such (Entropy)</th>
<th>Lower Bound on Energy of a Deep State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 1</td>
<td>2, 2</td>
<td>4!</td>
<td>2, 4</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4!</td>
<td>2, 2!</td>
</tr>
</tbody>
</table>
Construction of a potential
total stack size potential

\[ V_1(p_1, \ldots, p_n) = V_1(p_1) + \ldots + V_1(p_n) \]
\[ V_1(\Gamma \cdot p) = V_1(\Gamma i) = V_1(\Gamma) \]
\[ V_1(\Gamma(\vec{\Gamma}, a, q)) = V_1(\Gamma) + \epsilon_{\vec{a}} \]

\[ \vec{e} \cdot \tilde{\Gamma}(p) \]
Now we need to understand how these constrain the rate function. This is analogy on the local memory of the process of interest and the number of siblings, but this is an entirely local solution as the increasing reluctance to fork only depends and trivial cancellations, both of which leave.

\[ \Delta V_1 = (n - 1)V_1(\Gamma) \]  
\[ \Delta V_1 = m \varepsilon_{\tilde{a}} \]

For a fork, and by the simplicity property, the constraint translates into \( \log k \). Let us write

\[ \text{realize the ratio constraint as:} \]

\[ k_f^- = 1 \]
\[ k_f^+ = e^{-(n-1)V_1(\Gamma)} \]

\[ k_{\tilde{a}}^- = 1 \]
\[ k_{\tilde{a}}^+ = e^{-m \varepsilon_{\tilde{a}}} \]
Given a path $\gamma$ from $\emptyset \cdot p_0$ to $p$:

$$V_0(p) = \sum_{\tilde{a} \in A^*} \sum_{x \rightarrow s \cdot y \in \gamma} (-1)^{v(s)} \epsilon_{\tilde{a}}$$

ratio constraint:

$$k_f^- = k_f^+$$

$$k_{\tilde{a}}^- / k_{\tilde{a}}^+ = \exp(\epsilon_{\tilde{a}})$$
$V_1$ is truly concurrent, i.e. sensitive to sequential expansion.

$V_0 \leq V_1$

Potentially more divergent.

No matter how costly a synch, $V_0$ diverges.

What about $V_1$?
Lemma 5  For large $ns$, $\log |T(n)| \leq \beta_+ \alpha^2 O(n \log n)$

Lemma 4  Suppose $\beta_- > 1$, $\epsilon_m > 0$, $p \in \Sigma_n(p_0)$:

$$\frac{\epsilon_m}{\log 4 + \log(\beta_+ + 1)} \cdot n \log n \leq V_1(p)$$
Proposition 1  Suppose $1 < \beta_-$, and $\beta_+ \alpha^2 \log(4(\beta_+ + 1)) < \epsilon_m$, then:

$$Z(p_0) := \sum_{p \in \Omega(p_0)} e^{-V_1(p)} < +\infty$$
epilogue
Simulated annealing with "local" temperatures

The bounds are rough

Control growth rate?

What with irreversible actions?

Other potentials?

Work with general steady states?

Reactive modules? Something else than CCS

What kind of problem?

Obvious connexion with rewriting theory

\[ k_f^- = 1 \]
\[ k_f^+ = e^{-(n-1)V_1(\Gamma)} \]
idea III: energy-based programming/distributed Metropolis
Code = statics/potential + transition/moves + compatible kinetics

Energy as syntax
Self-organised energy-based dynamics
... stochastic machine learning

$$\arg \max \bar{\epsilon}. \sum_{p \in \partial X} \pi(\bar{\epsilon}, p) = \int 1_{\partial X} d\pi$$