q-bio 2018

Summer School

Matthew Fricke

Research Assistant Professor  Applications Scientist

UNM | Computer Science

Biological Computation Lab

CENTER FOR ADVANCED RESEARCH COMPUTING
Models

A model only makes sense in terms of some relation that is preserved.

• A model that makes *predictions* about some system
• A models used to define *computation*
• *Existence proof* models (models demonstrating the possibility of something).
• A model used to *explain* something that already happened.
Models

• `Now it would be very remarkable if any system existing in the real world could be exactly represented by any simple model. However, cunningly chosen parsimonious models often do provide remarkably useful approximations.`

• `For such a model there is no need to ask the question "Is the model true?". If "truth" is to be the "whole truth" the answer must be "No". The only question of interest is "Is the model illuminating and useful?".`

Models

• "Now it would be very remarkable if any system existing in the real world could be *exactly* represented by any simple model. However, cunningly chosen parsimonious models often do provide remarkably useful approximations."

• "For such a model there is no need to ask the question "Is the model true?". If "truth" is to be the "whole truth" the answer must be "No". The only question of interest is "Is the model illuminating and useful?"."

• “All models are wrong, some are useful.”

Models as Homomorphic Maps
Commutativity of the Diagram

Algorithm A

World at time t
Modeling Relation M
Model at time t

Laws L

World at time t + 1
Modeling Relation M
Model at time t + 1

M is an equivalence relation.
Model M is valid if this is a homomorphic map:

\[ M(L(x)) = A(M(x)) \]
Models as Homomorphic Maps

transformation of one set into another that preserves in the second set the relations between elements of the first.
Models

• “It can scarcely be denied that the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience.”

Attributed to Albert Einstein in “On the Method of Theoretical Physics,” the Herbert Spencer Lecture, Oxford, June 10, 1933. This is the Oxford University’ Press
Equivalence Classes

• Equivalence class =
  \[ \{x \mid x \in R\} \text{ and } R \text{ is an equivalence relation.} \]

• R is an equivalence relation:
  • Reflexive: \( \forall x (xRx) \)
  • Symmetric: \( xRy \Rightarrow yRx \)
  • Transitive: \( (xRy) \land (yRz) \Rightarrow (xRz) \)

  The relation does not change unless world changes, the relation is preserved between the model and world, the model and world stay consistent over time.

• Example: \( xRy \Leftrightarrow x \text{ and } y \text{ are in the same little box.} \)

Set of Objects

Partition set into 6 little boxes

Equivalence classes
Examples of Equivalence Relations

• “Is similar to" or "congruent to" on the set of all triangles.
• Logical equivalence of statements in logic.
• "Has the same image under a function" on the elements of the domain of the function.

What’s not an equivalence relation?
• The relation "≥" between real numbers is reflexive and transitive, but not symmetric. For example, 7 ≥ 5 does not imply that 5 ≥ 7. It is, however, a partial order.
• The relation "is a sibling of" on the set of all human beings is not an equivalence relation.
  • Is Symmetric (if A is a sibling of B, then B is a sibling of A)
  • Not reflexive (no one is a sibling of himself),
  • Not transitive (since if A is a sibling of B, then B is a sibling of A, but A is not a sibling of A).
Example Homomorphism:
Multiplication of Integers

• Model all pairs of integers and their product:
  • e.g., 14792 x 4183584 = 61883574528

• Model:
  • Even X Even = Even
  • Even X Odd  = Even
  • Odd X Even = Even
  • Odd X Odd = Odd
Example Homomorphism:  
Multiplication of Integers

Model:
Even \times Even = Even
Even \times Odd = Even
Odd \times Odd = Odd
Odd \times Even = Even

Model relationship:

\[ M(L(x)) = M(2n \times 2m = 2k) = Even \times Even = Even \]

The relationships are preserved under our model.

Model relationship:

\[ 2n \times 2m = 2k \times Even \times Even = Even \]
\[ 2n+1 \times 2m+1 = 2k+1 \times Odd \times Odd = Odd \]
\[ 2n \times 2m+1 = 2k \times Even \times Odd = Even \]
\[ 2n+1 \times 2m = 2k+1 \times Odd \times Even = Even \]
Lattice Gas Models (LGCA)

• Gasses and fluids can be modelled with continuous models
• That is, we can use continuous values of pressure, temperature, and velocity
Lattice Gas Models

• Gasses and fluids can be modelled with continuous models
• That is, we can use continuous values of pressure, temperature, and velocity
• What happens when we get to extreme cases:
  • If we are modelling a disk drive head moving just a micron above the platter these continuous models break down.
  • We have to model the individual molecules of gas.
Lattice Gas Models

• Gasses and fluids can be modelled with continuous models
• That is, we can use continuous values of pressure, temperature, and velocity
• What happens when we get to extreme cases:
  • If we are modelling a disk drive head moving just a micron above the platter these continuous models break down.
  • We have to model the individual molecules of gas.

• If we are modelling systems with very high energies (such as a nuclear explosion) we have to have a discrete model of the internal states of the atoms involved.
Lattice Gas Models

• If the molecules are very cold quantum effects start to dominate their interactions.
  • Here we have to model the quantum effects explicitly.
Lattice Gas Models

• If the molecules are very cold quantum effects start to dominate their interactions.
  • Here we have to model the quantum effects explicitly.

• These systems require models of the **microscopic** behaviour

• Models that are able to describe the behaviour of the system using just pressure, velocity, and temperature are **macroscopic**.

• Of course, we could model all gasses and fluids at the microscopic level.
Lattice Gas Models

• Cellular automata are used to model molecular systems.

• The use of cellular automata to model particles such as gasses, fluids, and

• The propagation of subatomic particles was pioneered by Stanislaw Ulam and John von Neumann in the 1950s.
(a) initial lattice  
(b) propagation  
(c) collision handling
Macrophage encounters an invader
Immune System Signaling (abridged)

Macrophage destroys the invader
Immune System Signaling (abridged)

Macrophage displays proteins from the invader
T-Cell are activated by the macrophage if their receptors match the displayed proteins.

we care about this…
B-Cells confirm that the proteins are from a pathogen encountered previously.
B-Cells change state and emit antibodies that bind to the invader.
Macrophages are recruited to the antibodies
Recruited macrophages destroy the invaders and display their proteins.

The cycle repeats and the immune response increases exponentially.
Foreign protein snippets are displayed by macrophage

Receptors that match particular protein snippets are displayed by T-cells.

In vivo, T cells are stimulated by monovalent binding to ligands on “antigen presenting cells” (e.g. Macrophages and B-Cells)
In vivo, T cells are stimulated by monovalent binding to ligands on “antigen presenting cells” (e.g. Macrophages and B-Cells)

The macrophage comes into contact with various T-cells.
In vivo, T cells are stimulated by monovalent binding to ligands on “antigen presenting cells” (e.g. Macrophages and B-Cells).

If the T-cell receptors match the protein snippet held by the ligands then they bind...

...and the ligands cluster together and the receptors cluster together.

How?
Our hypothesis: Cross Membrane binding between ligand-receptor pairs serves to combine the attractive forces between proteins in their own membranes. This would allow receptor or ligand groups that by themselves are do not cluster to “sum” the attractive forces and cluster.

Sounds simple but we can’t predict how strong the inter-membrane force needs to be in relation to the intra-membrane forces to cause phase separation. So model it!
Our Approach

- Write a model of phase separation on a single membrane
- Confirm that our results match those of previous phase transition models
- Implement two copies of the single membrane model and bring them into contact
- Add a cross-membrane binding force
- Under what circumstances do we get phase separation?
Metropolis Monte Carlo

- $n \times n$ toroidal lattice
- Each site on the lattice can hold a single protein
- At each discrete time-step all proteins choose a random direction to move
  - If the energy is reduced the motion is accepted.
  - Otherwise the motion is accepted with probability $e^{-\Delta E/kT}\varepsilon$.
- Repeat until we are confident that the system is in equilibrium

$\varepsilon = \text{Favorable contact energy (in kT) between neighboring proteins.}$
Measuring Phase Separation – Spatial Autocorrelation

- Autocorrelation Function $g(d)$
- Choose a protein and count the number of proteins at distance $d$ (then $\div 4$.)
The system **probabilistically** (Monte Carlo) enters a new lower energy configuration. The probability depends on how much the energy is decreased.

\[ P = e^{-\frac{\Delta E}{k_B T}} \]

Where the energy change is given by the binding energy \( N_{pp} \) at the proposed site, \( s_1 \) verses the current site \( s_0 \).

\[ \Delta E = -\varepsilon k_B T \left( N_{pp, s_1} - N_{pp, s_0} \right) \]
The system \textit{probabilistically} (Monte Carlo) enters a new lower energy configuration. The probability depends on how much the energy is decreased.

\[ P = e^{-\Delta E/k_B T} \]

\( k_B \) Boltzmann Constant \( T \) Temperature

Where the energy change is given by the binding energy \( N_{pp} \) at the proposed site, \( s_1 \) verses the current site \( s_0 \).

\[ \Delta E = -\varepsilon k_B T \left( N_{pp,s1} - N_{pp,s0} \right) \]

Not a dynamics model! The Monte Carlo model is sampling the space of possible protein configurations. The sites S0 and S1 could have been chosen randomly.
The system **probabilistically** (Monte Carlo) enters a new lower energy configuration. The probability depends on how much the energy is decreased.

\[
P = e^{-\Delta E/k_B T}
\]

\(k_B\) Boltzmann Constant \(T\) Temperature

Where the energy change is given by the binding energy \(N_{pp}\) at the proposed site, \(s_1\) verses the current site \(s_0\).

\[
\Delta E = -\varepsilon k_B T (N_{pp,s_1} - N_{pp,s_0})
\]

Not a dynamics model! The Monte Carlo model is sampling the space of possible protein configurations. The sites S0 and S1 could have been chosen randomly.

\[
\frac{P_{i\rightarrow j}}{P_{j\rightarrow i}} = \exp\left[-\frac{\Delta E_{ij}}{k_B T}\right]
\]

Microstate reversibility: Metropolis rule.
Predictions from Theory

The protein autocorrelation will scale with binding energy as:

\[ \xi \propto \frac{1}{\varepsilon_c - \varepsilon} \]
Predictions from Theory

The protein autocorrelation will scale with binding energy as:

\[ \xi \propto \frac{1}{\varepsilon_c - \varepsilon} \]

Time spent proteins spend bound (p) vs unbound (u):

\[ \frac{\bar{t}_u}{t_p} = \frac{1 - c}{c} e^{-\varepsilon_x} \quad \text{c is the concentration of proteins.} \]
Predictions from Theory

The effective interprotein interaction is:

$$\varepsilon_{\text{eff}} = \frac{\varepsilon t_u + 2 \varepsilon t_p}{t_u + t_p} = \frac{(1 - c)e^{-\varepsilon_x} + 2c}{(1 - c)e^{-\varepsilon_x} + c} \varepsilon$$
Proteins are randomly distributed (\(\varepsilon\) subcritical)

Proteins are highly correlated (\(\varepsilon\) supercritical)

Correlation Functions for Three Values of \(\varepsilon\)

- \(\varepsilon = 0\) (green line)
- \(\varepsilon = 0.6\) (red line)
- \(\varepsilon = 1.3\) (blue line)
Fit Correlation Functions to an Exponential \( y = C_1 e^{C_2 x} \)

(fit deteriorates as critical epsilon reached)
Fit exponential length constants to \( \frac{C_1}{\epsilon - \epsilon_c} + C_2 \)

This successful fit confirms that our model matches previous work on phase separation

Two Membrane Model
Density variance as a measure of phase separation

Calculating the autocorrelation, exponents, and critical exponents is too slow
Instead: calculate the protein density for all overlapping 3x3 squares on the lattice
Standard deviation is a measure of phase separation

Low $\sigma$, one phase

High $\sigma$, two phase
Complete phase separation occurs at 0.27

Random protein distributions have been observed to have values of between 0.09 and 0.105
Fig. 2. Fraction of proteins in clusters (dimers or higher aggregates), as a function of protein density, for various attractive interaction energies. The clustered fraction was fit to a background level of statistical aggregation, plus a mass action term (a dimerization equilibrium). The dimerization constant $K$ is plotted in the inset versus the binding energy, and shows the expected exponential dependence. Exact correspondence is not expected, because higher order aggregation is possible. Data from $100 \times 100$ lattice run for 10,000 iterations.

Fig. 3. Fraction of proteins in intermembrane dimers as a function of protein density in both membranes. At zero interaction energy, the fraction of dimers is the same as the protein density, as expected for random associations. The dimer concentration can be well fit by the sum of the background association, plus a mass action term. The dimerization constant for the mass action term is well fit to an exponential in binding energy, inset.
Fig. 4. (a) left: $100 \times 100$ double lattice at step 6500 with a protein concentration of 0.05 on each lattice. Intra-lattice protein interaction energy is $0.6 \ k_B T$ and the intermembrane interaction energy is zero. At this interaction energy, large clusters are never observed. (b) right: The same lattice configuration as in (a) but with an added intermembrane interaction energy of $5.0 \ k_B T$. Note the formation of large clusters as a consequence of the added intermembrane interaction.
Fig. 5. The reciprocal of the mean number of clusters (per membrane) in the two-membrane model, as a function of the intra- and intermembrane protein interaction energies. Each of the two 100×100 square lattices hosted 500 proteins, a density of 0.05. (Bottom) a 3D plot; (Top) same data, viewed from above. Color coding helps to identify the range of parameters that give strong clustering: blue and purple colors correspond to fewer than 2 clusters per membrane in the ensemble. The model was run for 15,000 iterations. To reduce the statistical variation, $1/N_c$ was averaged over 5 runs.

Fig. 6. The strengths of the intra-($\varepsilon$) and intermembrane ($\varepsilon_x$) protein interaction energies (in $k_B T$) required to give an effective interprotein interaction of 2 $k_B T$, according to a simple “mean field” estimate. 2 $k_B T$ is the threshold for receptor aggregation at this concentration (5%). The line should be compared with the boundary between aggregated (blue) and dispersed (red) protein phases in Fig. 5, Top.
RuleBuilder Layout

The Drawing Board is where containers, components, edges and operators can be placed in order to create the molecules, species, reaction rules, observables, and patterns that form a BioNetGen model.

Defined objects, such as Molecule Templates, Species, and Reaction Rules are displayed in separate windows.

Toolbar
Adding Containers and Components

A component is added by entering “Add Components” mode on the toolbar and left-clicking in the Drawing Board.
Renaming Components and Containers

Components and Containers can be renamed by left-clicking their label. Only alphanumeric characters and the underscore are allowed in labels.
Resizing Containers

Containers can be resized by dragging the “resize handles” that appear when they are selected.

Selected Containers always appear blue.
Creating Molecule Types

Molecules used in a model have to be defined and registered as a “Molecule Type” before they can be used in reaction rules and species.

To define a molecule type, add the desired components to the molecule, then right click (ctrl-click on Mac) on the container and select Create Molecule Type.
Setting Allowed Component States

Components may take on different states to indicate conformation or covalent modification, such as phosphorylation.

Selecting Create Molecule Type brings up the Component State dialog. Add an allowed state by typing in the Add Allowed State box and clicking Add. Components don't need to have any allowed states. Exit the dialog by clicking Done.
Identifying Valid and Invalid Molecules

Containers matching valid types are **green**.

Containers not matching a valid type are **red**.

Dashed line indicates an incomplete match.
Copying Objects with the Selection Box

Draw a box around objects on the Drawing Board to select them. Partially enclosed objects are not selected.
Copy objects with the selection box:

Clicking and dragging the selection box moves everything selected; holding down the shift key while dragging copies the selection to a new location.
Creating a Reaction Rule

Reaction rules are created by arranging containers and operators to construct a formula for the reaction.

The '+' operator separates reactants or products in a list.

The arrow operator separates reactants and products.
Creating a Reaction Rule

The type of arrow determines whether a reaction is reversible or irreversible.
Defining Products

Use Add Edges to create a bond between the components
Defining Products

Create the bond by clicking on the two components to be linked.
Creating the Rule

Draw a selection box enclosing the rule, right-click (ctrl-click on Mac), and select Make Rule.

If Make Rule is grayed out, make sure all of the objects are valid and the box is enclosing all of the elements.
Make Rule Dialog

Set Rule Name, rate constants, and optional annotation in the dialog box.

For a parameter being used for the first time, set a numerical value in the Rate box.
Reaction Rules Window

Rule now appears in the Reaction Rules Window.
Defining Seed Species

The network is defined by applying the reaction rules to a set of seed species.

Draw a selection box around a connected set of molecules and right-click (ctrl-click on Mac) to define a species.

All items in selection box should have solid green lines, indicating the the molecules are fully defined.
Species Dialog Box

A dialog box appears for setting the name and initial concentration of the species.
Seed Species Window

New species appears in the Seed Species Window.
Defining Observables

Observables are concentration sums over species with particular properties and correspond to model outputs, such as total phosphorylation of a protein.

Draw a selection box around the pattern defining the Observable, right click (ctrl-click on Mac) and select Make Observable.

If this option is not available, make sure that all molecules in the observable are defined.
Make Observables Dialog

Set Rule Name and Type.

Type **Molecules** weights the concentration of each matching species by the number of times the defined pattern matches the species. Use this for quantities like total phosphorylation of a site on a protein or total number of receptors in aggregates.

Type **Species** gives unit weight to the concentration of each matching species. Use this type to get the concentration of complexes of a particular type.
If the Observables Window is not visible, Select Observables Palette from the View menu.
Observables Window
Running the Model

Once Reaction Rules, Seed Species, and Observables (optional) have been defined, the model can be simulated by pressing Run BioNetGen button.
BioNetGen Engine Settings

Use the Browse buttons to set paths for the BioNetGen Engine and the Work file.
These settings can also be changed by selecting Settings under the File menu.
The SimConfig Panel

This panel can also be accessed on the tool bar.

The Reaction Network Generation and Simulation Settings Dialog appears the first time a simulation is run. Use these options to control how the simulation will be run.
The SimConfig Panel

The size of complexes and the reaction network can be limited by setting maximum values for the stoichiometry of molecules.
The SimConfig Panel

A portion of the network may be preequilibrated by selecting “run equilibrate” and a subset of defined seed species.
The total simulation time and the times at which concentrations are sampled are set here.
The SimConfig Panel

Either a stochastic (SSA) or a deterministic (ODE) simulation method may be chosen.
The SimConfig Panel

SBML output may be selected for export of the generated network to other applications.
Running the simulation brings up the BioNetGen Output Log or Log Window.

The Log Window displays the output of BioNetGen.
Plotting the Results

Once the simulation is finished, selecting the Plot Results button brings up a plotting window for the simulation results.
Plotting the Results

Either observables or species may be plotted.

Available Observables or Species are shown here.
Plotting the Results

A plot of the defined observable.
Thymus
Ultimate Naïve T cell source
Peripheral Tissue
Drains from lymphatic channels
Infected Cell
Naïve T cell
Activated T cell
Dendritic Cell
Antigen
Blood Vessels
Drain to other lymph nodes and blood vessels
Balancing Extent and Intensity for Target Detection Error
Thymus
Ultimate Naïve T cell source
Peripheral Tissue drains from lymphatic channels
Infected Cell
Naïve T cell
Activated T cell
Dendritic Cell
Blood Vessels
Drain to other lymph nodes and blood vessels
Balancing Extent and Intensity for Target Detection Error
Rate Limiting Step
Limiting Step
Naïve T cell source
The Needle in a Haystack

Lymph nodes have a volume $10^6$ times that of T cells.

100k T cells and 100k DCs. Small set of these are cognate.

T cells move at an average speed of 0.11 μm/s.

T cells searching systematically (raster scan) would discover an antigen target in 6 days on average.

Simple random walkers (Brownian) have an expected 30% success rate after 3 days.[1]

T cells are able to find cognate antigen in 3-8 hours and give up after 12-24 hours.

Background: Intensive vs Extensive Search

• We can describe any stochastic search pattern with distributions of vector lengths and turning angles.
Background: Intensive vs Extensive Search

• We can describe any stochastic search pattern with distributions of vector lengths and turning angles.
• Intensive searchers have lower displacement but search more thoroughly.
• Simple random search (Brownian motion) will eventually cover the entire area.
Background: Intensive vs Extensive Search

- We can describe any stochastic search pattern with distributions of vector lengths and turning angles.
- *Intensive* searchers have lower displacement but search more thoroughly.
- Simple random search (Brownian motion) will eventually cover the entire area.

- *Extensive* searchers cover more ground but leave gaps.
- Mean Squared Displacement (MSD) is a measure of search extent.
Two-photon microscopy

131 ex vivo observations
25,000 T cells tracked
Half hour observations
Extracting Tracks from Fluorescence in collaboration with the Cannon lab
Building a statistical model of T cell Search
Displacement: Distance travelled from starting position.
Simple random walk
Would have a log transformed MSD with slope $\alpha = 1$ and $\mathcal{H} = 2$.

We see a slope of $\alpha = 1.41$
Superdiffusion.

Fractal dimension = 1.41
(This is our measure of the intensity-extent trade-off)
Modelling T cell Search: Fit Step Lengths

- Observed
- Power Law
- Maxwell

Maximum-likelihood fitting [1]

Modelling T cell Search: Fit Step Lengths

- Observed
- Power Law
- Maxwell
- Lognormal

T cell Search in the Lung

- The lognormal CDF is still a good fit.
- Exponential is also good.
- This pattern of movement is somewhat less intensive than in lymph nodes.

Paulus Mrass et al., "ROCK regulates the intermittent mode of interstitial T cell migration in inflamed lungs," *Nature Communications*, 2017
Modelling T cell Search: Angle Correlation (Search in Lungs and T cells look similar)

Measure of correlation with previous direction. Correlated random walk?
Comparison of search efficiency with empirical observations

**Extensive search is better**

![Box plot showing efficiency comparison for unique contacts with intensity vs extent tradeoff: Target Detection Error.]

**Intense search is better**

![Box plot showing efficiency comparison for total contacts with intensity vs extent tradeoff: Target Detection Error.]

- Intensive
- Extensive

Intensity vs Extent Tradeoff: Target Detection Error
Comparison of search efficiency with empirical observations

Intensity vs Extent Tradeoff: Target Detection Error

Intensive search is better

Extensive search is better

Δ Efficiency (%) vs Unique Contacts

Δ Efficiency (%) vs Total Contacts

Brownian | Bootstrap | LogMCRW | Power-Law

Intensive | Extensive

Intensive search is better

Extensive search is better

Intensive | Extensive

Intensity vs Extent Tradeoff: Target Detection Error
Comparison of search efficiency with empirical observations

Extensive search is better

Intense search is better

Intensity vs Extent Tradeoff: Target Detection Error

Unique Contacts

Total Contacts
Comparison of search efficiency with empirical observations

**Extensive search is better**

Intensity vs Extent Tradeoff: Target Detection Error

- **Unique Contacts**
  - Intensive
  - Extensive

- **Total Contacts**
  - Intensive
  - Extensive

**Intense search is better**

- **Brownian**
- **Bootstrap**
- **LogMCRW**
- **Power-Law**

Δ Efficiency (%)
Why don’t T cells use the parameters that result in the most unique contacts?
Why don’t T cells use the parameters that result in the most unique contacts?

T cells have to revisit antigen multiple times for ligand and receptor rafts to form and signal to be properly integrated. [1,2]

Why don’t T cells use the parameters that result in the most unique contacts?

T cells have to revisit antigen multiple times for ligand and receptor rafts to form and signal to be properly integrated. [1,2]

As rarity (expected distance between targets and searcher) increases extensive search does better [3]. Intensive search does better when cognate antigen is common. T cells are able to take advantage of both.

In other work: Beyond Random Walks

Fibroblastic Reticulum Cells (FRC)

There are theories that depend on T cell associations with FRC Network and HEVs [1,2,3]

No signs of chemical attraction Between T cells and DCs contrary to [4].

T cell Search Summary
• T cells are superdiffusive when searching lymph nodes and lung tissue, helping to explain their efficiency.
T cell Search Summary

• T cells are superdiffusive when searching lymph nodes and lung tissue, helping to explain their efficiency.
• The search pattern in lymph nodes allows for signal integration (less important in lungs).
T cell Search Summary

• T cells are superdiffusive when searching lymph nodes and lung tissue, helping to explain their efficiency.
• The search pattern in lymph nodes allows for signal integration (less important in lungs).
• Associated with the FRC small world network which may also increase efficiency.
T cell Search Summary

- T cells are superdiffusive when searching lymph nodes and lung tissue, helping to explain their efficiency.
- The search pattern in lymph nodes allows for signal integration (less important in lungs).
- Associated with the FRC small world network which may also increase efficiency.
- Little spatial correlation with DCs which argues against DC recruitment.