Complexes and rules in Moleculizer

Larry Lok

The Molecular Sciences Institute
Berkeley, California
lok@molsci.org
June 14, 2007

Questions

1. What biological features and questions motivated the development of Moleculizer?

2. What capabilities does Moleculizer provide?

3. What are specific examples where using Moleculizer led to new insights?

4. What capabilities still need to be developed?

5. What are the essential entities and processes that you feel must be represented in any exchange format to support Moleculizer and your modeling needs?

What is Moleculizer?

1. Rule-based "mixed bag" stochastic simulator for cellular chemical reaction networks.

2. On-the-fly reaction network generation.

3. Event-queue-based form of Gillespie's first-reaction method.

4. New compartmental version: Cpt or "Compartmentalizer."

5. Cpt adapts Gillespie's direct method to interlace diffusion and reactions. Diffusion is *not* done with pseudo-reactions.

Motivations for developing Moleculizer

- I originally wrote Moleculizer for scientists at MSI to study the mating pheromone response pathway in yeast.
- Combinatorial explosion in hand-written models prompted the rule-based effort.
- In 2001, we were convinced that stochastic simulation was necessary (Elowitz). Gillespie's first-reaction method encouraged the on-the-fly approach. (MSI is not so stochastic now.)
- I was exploring shape theory, and using more detailed physical descriptions of proteins/complexes to influence simulated reaction kinetics, loosely *allostery,* seemed to fit the research interests of some MSI scientists (Yu).

Moleculizer capabilities, features, peculiarities

1. Also does generate-first method, as well as intermediates between generate-first and on-the-fly.

2. Reporting facilities; i.e. expressing what populations to track in output. This is like giving a rule specifying the reactant in a unary reaction.

3. Reaction network can of course be dumped, translated into SBML(L2) and a few "in-house" formats.

4. Feature: Structural classes "pre-filtered" for rule applicability.

5. Peculiarity: Implementation of allostery.

Feature: Structural classes "pre-filtered" for rule applicability

Moleculizer/Cpt's database of species has two levels:

1. Structure – how the mols ($\equiv$ species type?) connect with each other in the complex.

2. Modifications – how, where the mols are phosphorylated, ubiquitinylated, etc. Modifications have no internal structure, so that they are easier to examine, compare, etc.

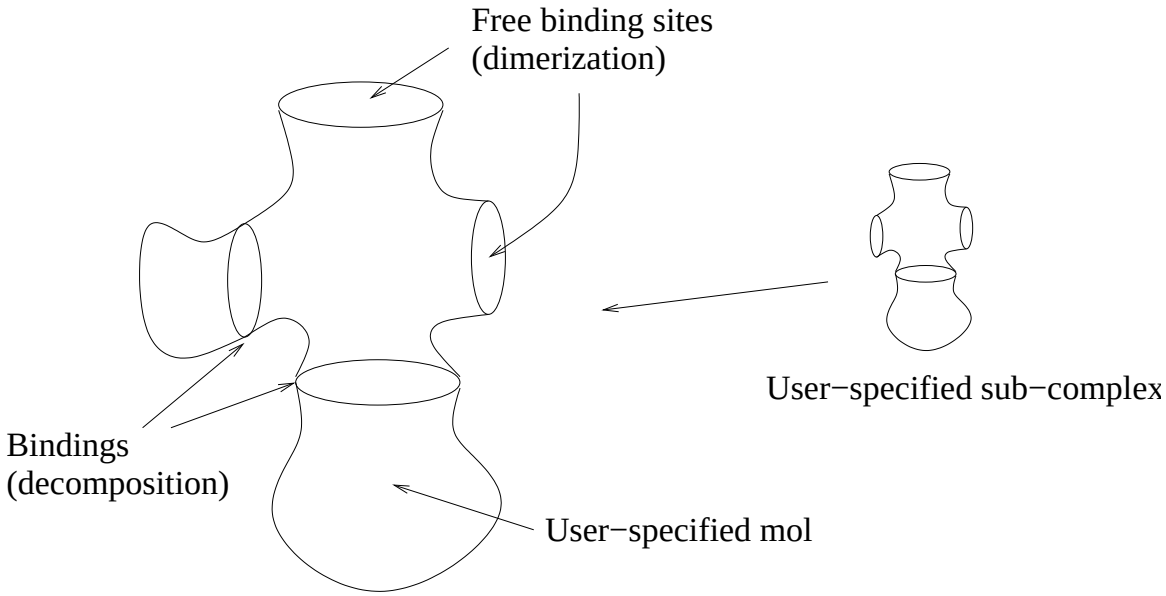
Feature: Structural classes "pre-filtered" for rule applicability

The rules that are applicable to a complex are largely determined by its structure. When a particular structure is recognized, the rules that might apply to species with that structure become known.

Hence, a new species with the given structure need not go through the whole process again: rules need only test the new species's modifications.

(In retrospect: the determination of which rules apply to a new species did not need acceleration.)

Feature: Structural classes "pre-filtered" for rule applicability



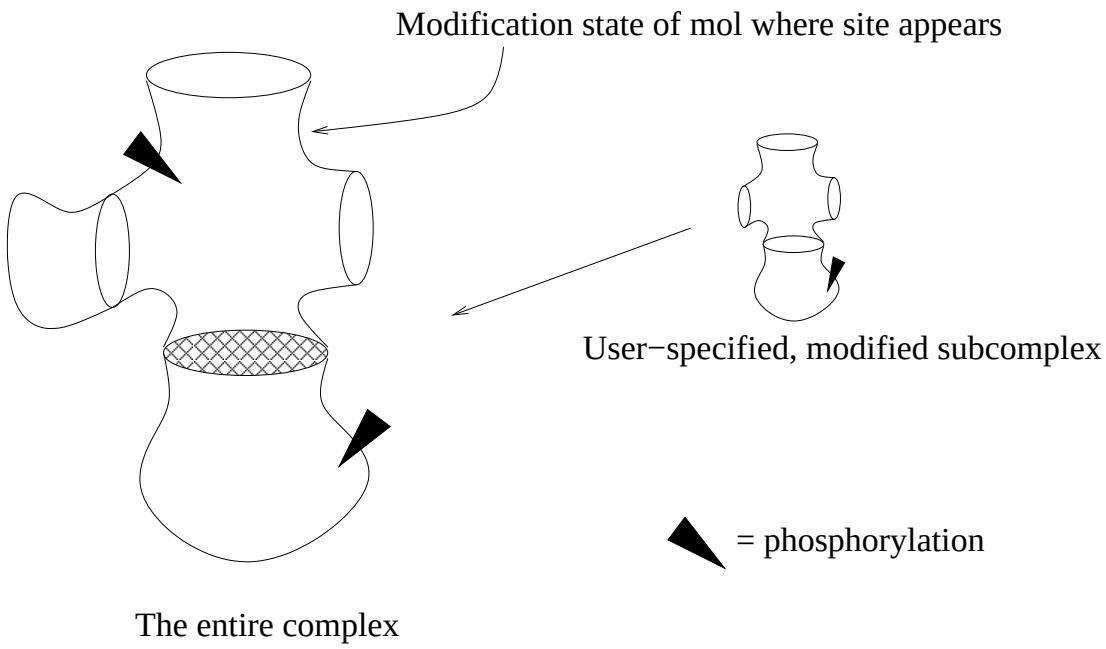
Different kinds of rules were to be triggered by different kinds of structural features. Therefore, I was inclined to make the specification of each kind of rule reflect the structural feature that triggered it. The idea of a really generic reaction generator didn't occur to me at the time.

Peculiarity: Implementation of allostery

Uses abstract binding site "shapes" to implement allosteric effects.

This presently involves no representation of the actual geometric/charge shape, just a label, usually the reason for the allosteric change of shape from default. E.g. `doubly_phosphorylated_shape`.

Peculiarity: Implementation of allostery



Sources of allosteric changes in site shape, hence binding kinetics.

Issue in BNG(?) and Moleculizer: resolving multiple allosteric effects. How does one know which of several applicable templates "wins"?

Example biological insights from using Moleculizer, Cpt

1. (Mzr) That, assuming fixed dissociation constants between kinases and Ste5 scaffold protein in mating pheromone response pathway, on- and off-rates affect signal transmission in a "bimodal" way, similar to Ste5 concentrations (ref?).

2. (Cpt) That we would not be able to tell whether oligomerization of Ste5-YFP (at the membrane, after initiation of signaling) would yield visible "clumps" or not without both oligomerization rates and rate of diffusion in the membrane (different from diffusion in the cytosol).

Some software insights from using Moleculizer, Cpt

1. Even when documented by html-based help and prompted by syntax-directed editing with a fairly good editor, XML is not popular with biologist users.

2. Early development based on the "in-house" software concept that frequent code modifications would be common, in particular "custom" reaction generators. Facilitating swapping these out, adding new ones, etc. was an early priority.

   Better to fix capabilities, then make them all available as simply as possible.

Capabilities still needed in Moleculizer, Cpt

1. Template-based reaction rule specification.

2. Species and reactions in Cpt exist/occur in one compartment at a time.

3. Misc: events triggered by simulation state. E.g. stop the simulation when species X reaches a certain population.

4. Misc: events read from a file. E.g. reset the population of alpha mating pheromone every simulated second with a value read from an input file.

5. Speculative: incorporate interesting biophysics to try to extrapolate rates of binding and unbinding for intra-molecular binding-site pairs. To put it another way, make some sort of progress on cooperative binding, intra-molecular stress on bindings, etc.

Exchange format requirements

What's "required" has more to do with what parts of Moleculizer/Cpt's functionality should be covered:

- If "allostery" concept is covered, then it would have major requirements just for specification. It would also gum up the works of template-based reaction rule specification.

- If "allostery" concept is not covered, say, because dropped from Moleculizer, then the template system in the Blinov et al. proposal seems like it should be fine.

Does SBML address "unknown requirements?" Does SBML have (or need) a strategy for coping with routine individualism and weirdness in client programs?

Miscellany

1. Template-based rules like Scheme macros vs. Common Lisp macros. Template-based macros are much easier to use and lead to fewer unexpected results, though they are not absolutely general.

2. SimTK.org, a Stanford-based biophysical modeling group is currently doing the sort of modeling that I thought might lead to something with allostery.

   E.g. see https://simtk.org/home/alphamol.

Thanks!

- To you for your kind attention.
- To the workshop organizers for bringing us together.
- To NIH and DARPA for support at MSI.
- To the MSI researchers who used Moleculizer, giving helpful feedback for its development.