LA-UR-17-28151
Approved for public release; distribution is unlimited.

Title: TRuML: a translator for rule-based modeling languages

Author(s): Suderman, Ryan T.
Hlavacek, William Scott

Intended for: ACM-BCB, 2017-08-21/2017-08-23 (Cambridge, Massachusetts, United States)

Issued: 2017-09-11

Disclaimer:
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

TRuML: A translator for rule-based modeling languages

Ryan Suderman, William S. Hlavacek

Los Alamos NATIONAL LABORATORY EST. 1943

Center for Nonlinear Studies

Dynamical systems biology

Modeling protein interaction networks traditionally done with ODEs or reaction networks

Two prominent issues:

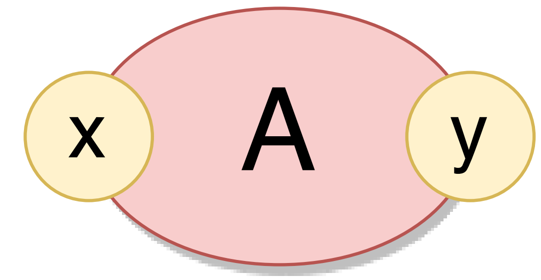
- Encoding (knowledge representation)
- Complexity

Los Alamos NATIONAL LABORATORY EST. 1943

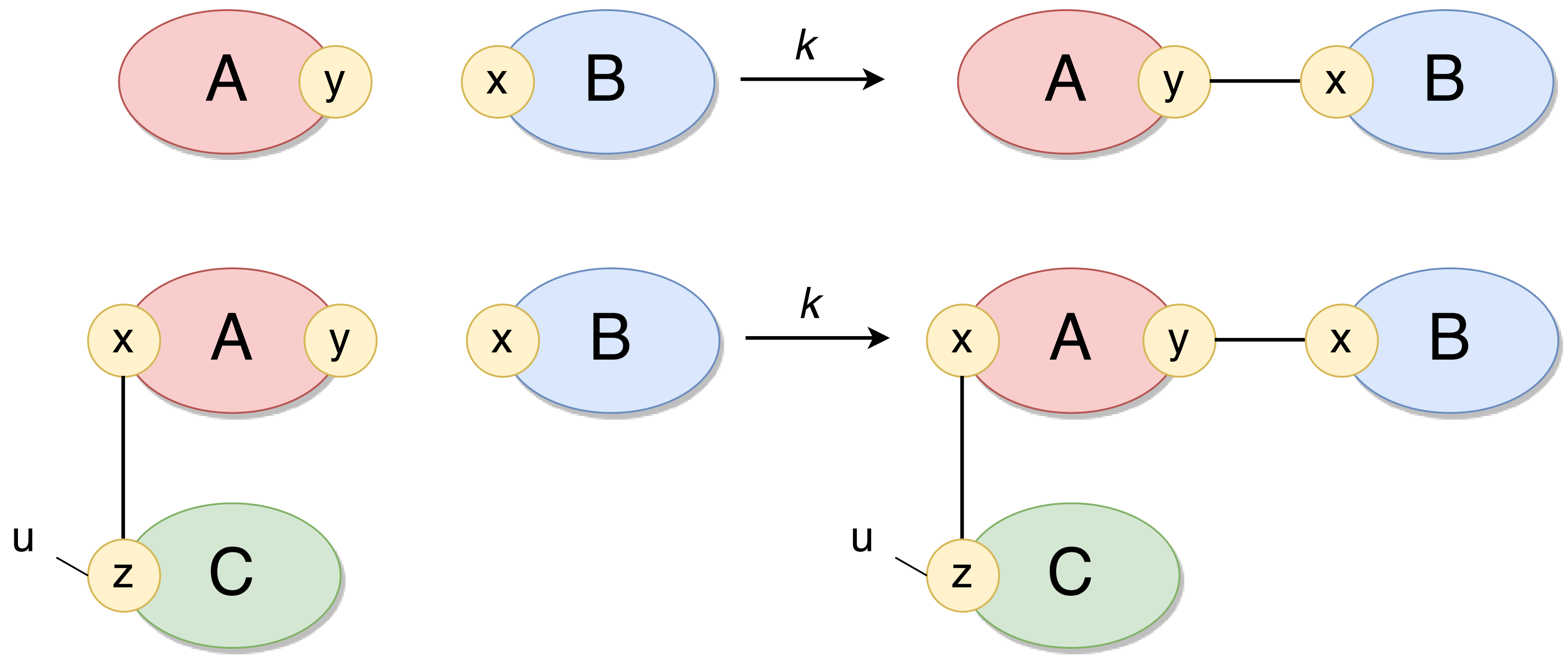
Center for Nonlinear Studies

Rule-based modeling

"Site graphs" represent molecules/complexes



Graph-rewriting rules represent sets of reactions



Chylek, L. A., et al, (2014), WIREs: Sys Biol Med

Danos, V., et al, (2007), Concur 2007

Los Alamos NATIONAL LABORATORY EST.1943

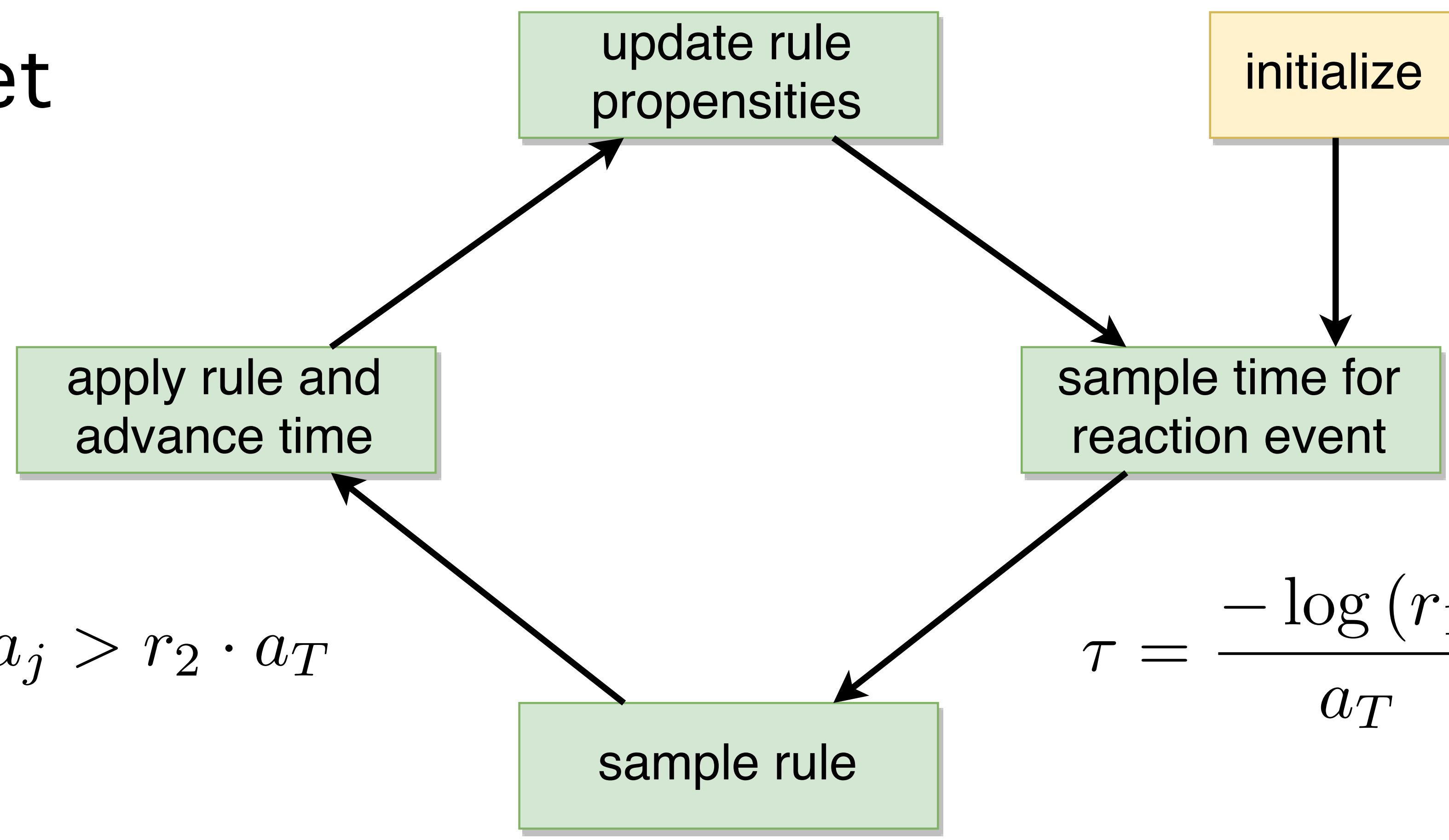
Center for Nonlinear Studies

Rule-based modeling

BioNetGen (BNGL) and Kappa languages

Both have “direct” KMC-based simulation engines

Each has a unique set of analysis tools



$$\sum_{j=1}^{i} a_j > r_2 \cdot a_T$$

$$\tau = \frac{-\log(r_1)}{a_T}$$

Danos, V., et al, (2007), LNCS

Gillespie, D. T. (2007). Ann Rev Phys Chem

Sneddon, M. W. et al, (2011), Nat Meth

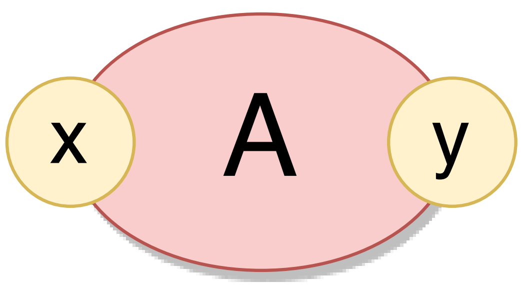
a_T - sum of all rules’ propensities

a_i - the propensity of rule i

Translation

TRuML is a tool for translating between Kappa and BNGL

Some model components can be trivially translated:



BNGL: `A(x,y)`

Kappa: `%agent: A(x,y)`

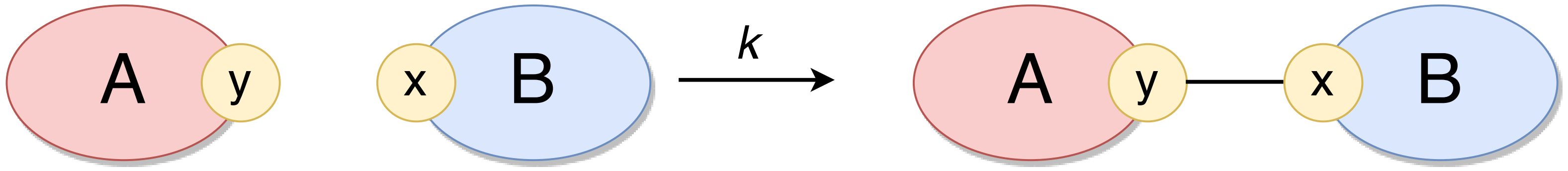
Others require syntactic modification:

BNGL: `x = log10(y + 1) / z`

Kappa: `%var: 'x' ([log]('y' + 1) / [log](10)) / 'z'`

Translation

Rules are similar syntactically, but with key differences:



Kappa:

```
A(y),B(x) -> A(y!1),B(x!1) @ k
```

BNGL:

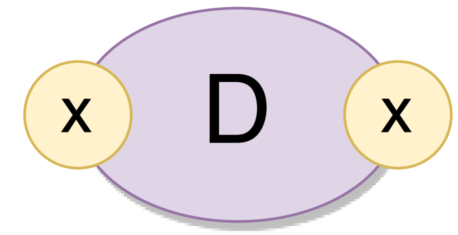
```
A(y) + B(x) -> A(y!1).B(x!1) k
```

Los Alamos NATIONAL LABORATORY EST. 1943

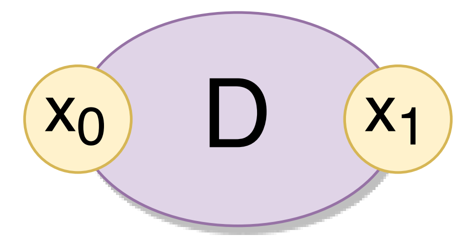
Center for Nonlinear Studies

Translating identically named sites

BNGL allows molecules with identical sites



Kappa's formalism requires distinct site names



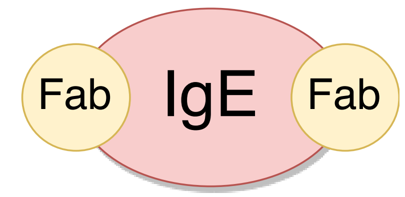
BNGL patterns involving identical sites must be expanded to accommodate Kappa's site naming conventions

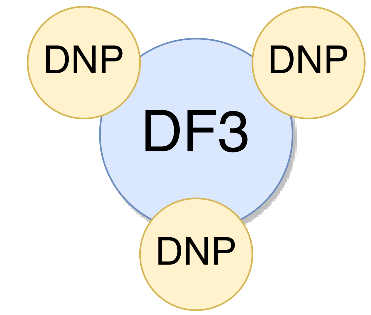
Los Alamos NATIONAL LABORATORY EST. 1943

Center for Nonlinear Studies

Translating identically named sites

Consider the immune response





Two types of binding rules:

- Free DF3 binding IgE
- Bound DF3 crosslinking 2 IgEs

BNGL:

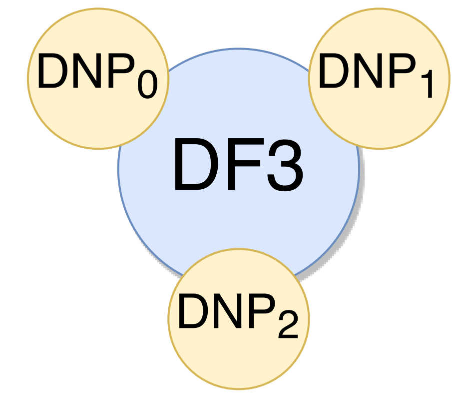
```
IgE(Fab)+DF3(DNP,DNP,DNP) -> IgE(Fab!1).DF3(DNP!1,DNP,DNP) k1
   IgE(Fab)+DF3(DNP,DNP!+) -> IgE(Fab!1).DF3(DNP!1,DNP!+) k2
```

Los Alamos NATIONAL LABORATORY EST.1943

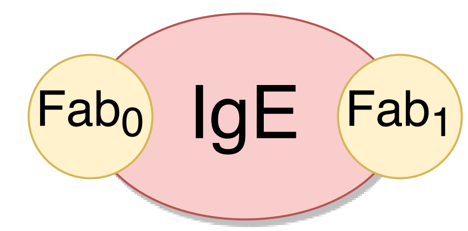
Center for Nonlinear Studies

Translating identically named sites

First, the molecule types' sites must be renamed



Patterns containing these molecule types must be combinatorially expanded




```
IgE(Fab!1).DF3(DNP!1,DNP,DNP)
```

```
IgE(Fab0!1),DF3(DNP0!1,DNP1,DNP2)
IgE(Fab1!1),DF3(DNP0,DNP1!1,DNP2)
IgE(Fab0!1),DF3(DNP0,DNP1,DNP2!1)
IgE(Fab1!1),DF3(DNP0!1,DNP1,DNP2)
IgE(Fab0!1),DF3(DNP0,DNP1!1,DNP2)
IgE(Fab1!1),DF3(DNP0,DNP1,DNP2!1)
```

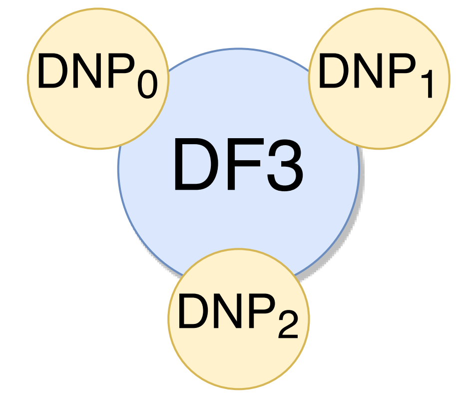
Los Alamos NATIONAL LABORATORY EST. 1943

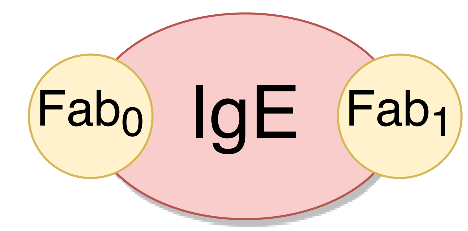
Center for Nonlinear Studies

Translating identically named sites

This is not sufficient for certain cases

Consider the crosslinking rule's DF3 reactant:



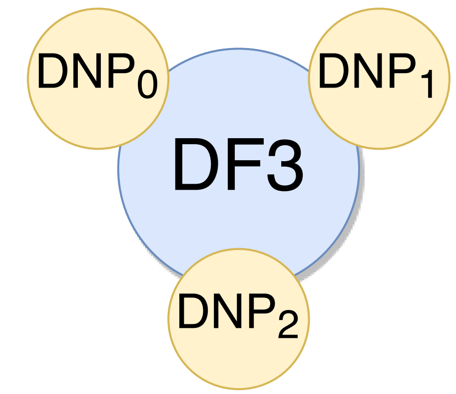


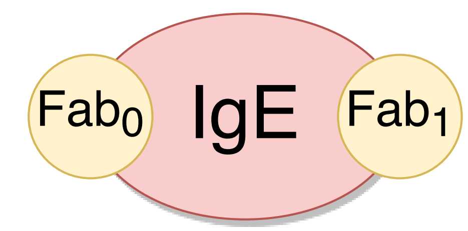

```
DF3(DNP,DNP!+)
```

```
DF3(DNP0,DNP1!_)
DF3(DNP0,DNP2!_)
DF3(DNP1,DNP0!_)
DF3(DNP1,DNP2!_)
DF3(DNP2,DNP0!_)
DF3(DNP2,DNP1!_)
```

Translating identically named sites

```
DF3(DNP0,DNP1!_)
DF3(DNP0,DNP2!_)   →   DF3(DNP0,DNP1!_,DNP2!_)
DF3(DNP1,DNP0!_)
DF3(DNP1,DNP2!_)
DF3(DNP2,DNP0!_)
DF3(DNP2,DNP1!_)
```





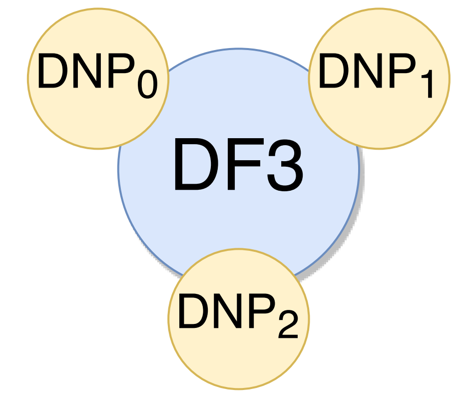
Overlapping patterns cause an overestimate of a rule’s propensity

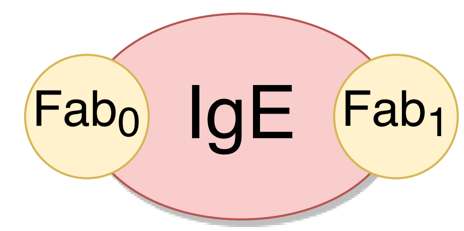
Additional context is needed

Translating identically named sites

Generally, if multiple identical sites exist and are underspecified in a pattern:

0. Perform combinatorial expansion as before





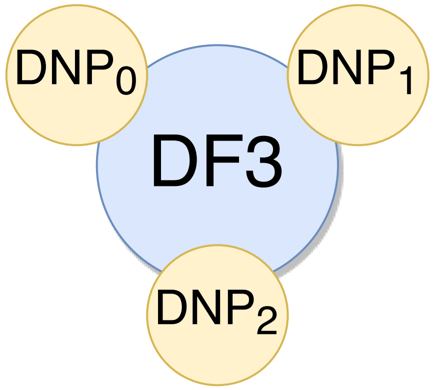

```
DF3(DNP,DNP!+)  →  DF3(DNP0,DNP1!_)
                   DF3(DNP0,DNP2!_)
                   DF3(DNP1,DNP0!_)
                   DF3(DNP1,DNP2!_)
                   DF3(DNP2,DNP0!_)
                   DF3(DNP2,DNP1!_)
```

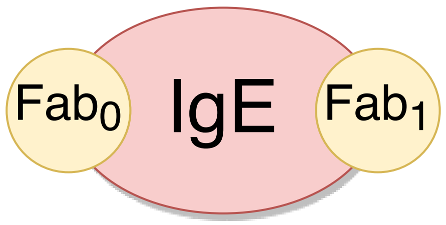
Translating identically named sites

Generally, if multiple identical sites exist and are underspecified in a pattern:

0. Perform combinatorial expansion as before

1. Determine all possible states for the site in question

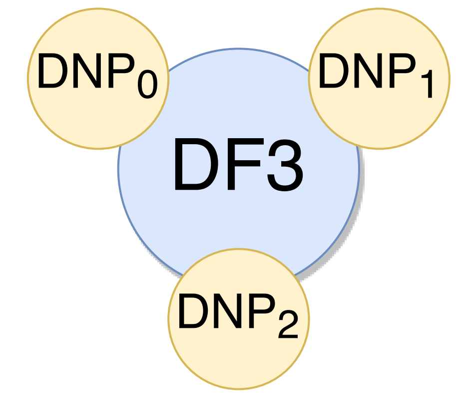





```
DNPN   →   DNPN
           DNPN!_
```

Translating identically named sites

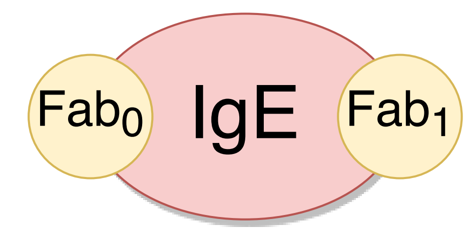
Generally, if multiple identical sites exist and are underspecified in a pattern:



0. Perform combinatorial expansion as before

1. Determine all possible states for the site in question

2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion



all sites — specified sites — unspecified sites

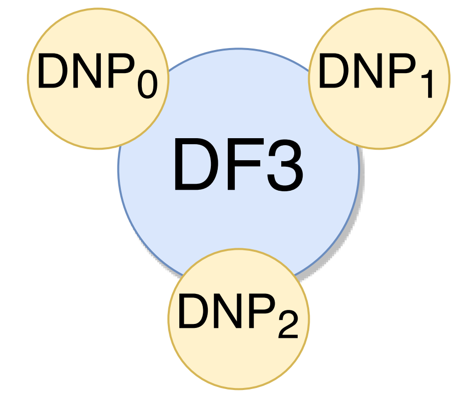
```
DF3(DNP0,DNP1!_) ⟶ {DNP0,DNP1,DNP2} - {DNP0,DNP1} = {DNP2}

{DNP2} X {DNPN, DNPN!_} = {DNP2, DNP2!_}
```

Los Alamos NATIONAL LABORATORY EST. 1943

Center for Nonlinear Studies

Translating identically named sites

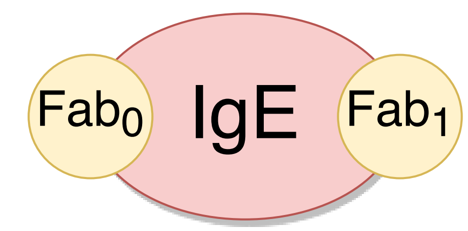
Generally, if multiple identical sites exist and are underspecified in a pattern:



0. Perform combinatorial expansion as before

1. Determine all possible states for the site in question

2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion

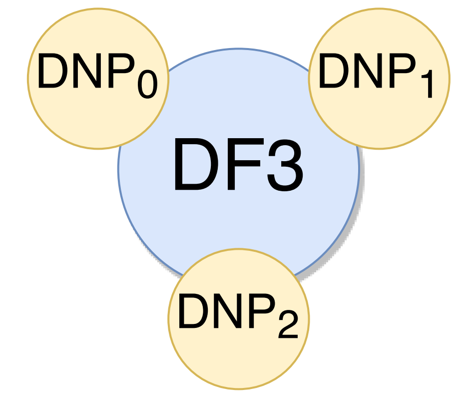


3. Generate new patterns by adding all unspecified site combinations to each pattern in the expansion

```
DF3(DNP0,DNP1!_) + {DNP2, DNP2!_} ⟶

        {DF3(DNP0,DNP1!_,DNP2), DF3(DNP0,DNP1!_,DNP2!_)}
```

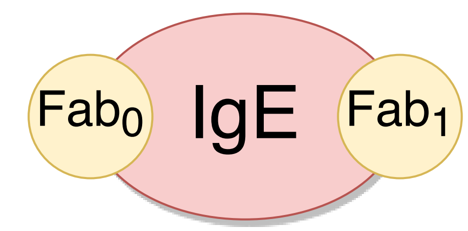
Translating identically named sites

Generally, if multiple identical sites exist and are underspecified in a pattern:



0. Perform combinatorial expansion as before

1. Determine all possible states for the site in question



2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion

3. Generate new patterns by adding all unspecified site combinations to each pattern in the expansion

4. Prune identical patterns from list

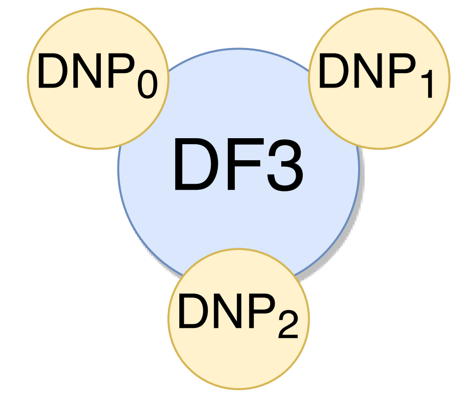
```
                   DF3(DNP0,DNP1!_)      DF3(DNP0,DNP1!_,DNP2)
                   DF3(DNP0,DNP2!_)      DF3(DNP0,DNP1!_,DNP2!_)
                   DF3(DNP1,DNP0!_)      DF3(DNP0,DNP2!_,DNP1)
DF3(DNP,DNP!+) →   DF3(DNP1,DNP2!_)  →   DF3(DNP1,DNP0!_,DNP2!_)
                   DF3(DNP2,DNP0!_)      DF3(DNP1,DNP0!_,DNP2)
                   DF3(DNP2,DNP1!_)      DF3(DNP2,DNP0!_,DNP1!_)
```

Los Alamos NATIONAL LABORATORY EST. 1943

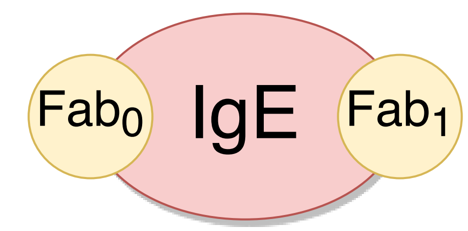
Center for Nonlinear Studies

Translating identically named sites

Generally, if multiple identical sites exist and are underspecified in a pattern:

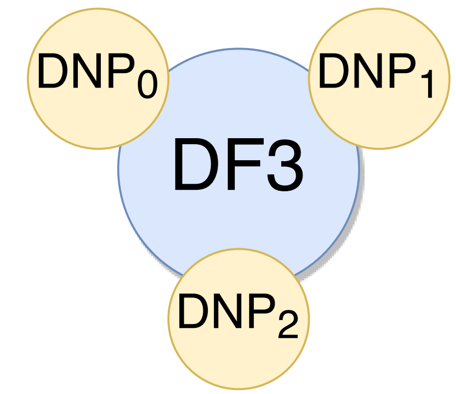


0. Perform combinatorial expansion as before
1. Determine all possible states for the site in question
2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion
3. Generate new patterns by adding all unspecified site combinations to each pattern in the expansion
4. Prune identical patterns from list



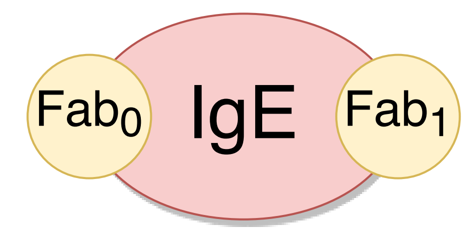

```
                    DF3(DNP0,DNP1!_)      DF3(DNP0,DNP1!_,DNP2)
                    DF3(DNP0,DNP2!_)      DF3(DNP0,DNP1!_,DNP2!_)
                    DF3(DNP1,DNP0!_)      DF3(DNP0,DNP2!_,DNP1)
DF3(DNP,DNP!+)  →   DF3(DNP1,DNP2!_)  →   DF3(DNP1,DNP0!_,DNP2!_)
                    DF3(DNP2,DNP0!_)      DF3(DNP1,DNP0!_,DNP2)
                    DF3(DNP2,DNP1!_)      DF3(DNP2,DNP0!_,DNP1!_)
```

Translating identically named sites



Generally, if multiple identical sites exist and are underspecified in a pattern:

0. Perform combinatorial expansion as before
1. Determine all possible states for the site in question
2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion
3. Generate new patterns by adding all unspecified site combinations to each pattern in the expansion
4. Prune identical patterns from list




```
DF3(DNP,DNP!+)
```

→

```
DF3(DNP0,DNP1!_)
DF3(DNP0,DNP2!_)
DF3(DNP1,DNP0!_)
DF3(DNP1,DNP2!_)
DF3(DNP2,DNP0!_)
DF3(DNP2,DNP1!_)
```

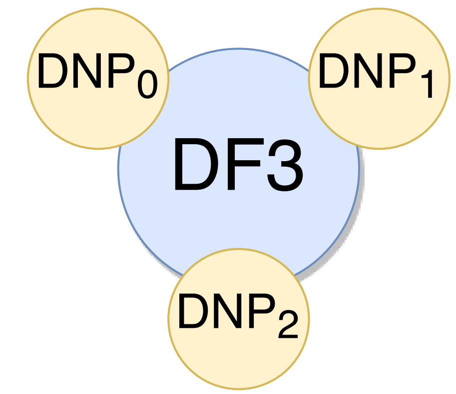
→

```
DF3(DNP0,DNP1!_,DNP2)
DF3(DNP0,DNP1!_,DNP2!_)
DF3(DNP0,DNP2!_,DNP1)
DF3(DNP1,DNP0!_,DNP2!_)
DF3(DNP1,DNP0!_,DNP2)
DF3(DNP2,DNP0!_,DNP1!_)
```

Los Alamos NATIONAL LABORATORY EST.1943

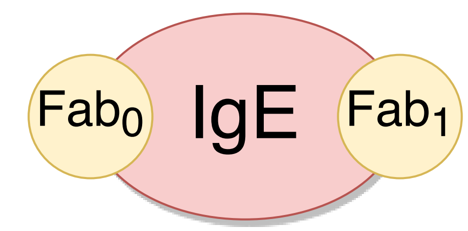
Center for Nonlinear Studies

Translating identically named sites



Generally, if multiple identical sites exist and are underspecified in a pattern:

0. Perform combinatorial expansion as before
1. Determine all possible states for the site in question
2. Take product of possible states and unspecified Kappa site names for each pattern in the expansion
3. Generate new patterns by adding all unspecified site combinations to each pattern in the expansion
4. Prune identical patterns from list




```
                   DF3(DNP0,DNP1!_)     DF3(DNP0,DNP1!_,DNP2)
                   DF3(DNP0,DNP2!_)     DF3(DNP0,DNP1!_,DNP2!_)
                   DF3(DNP1,DNP0!_)     DF3(DNP0,DNP2!_,DNP1)
DF3(DNP,DNP!+) ->  DF3(DNP1,DNP2!_)  -> DF3(DNP1,DNP0!_,DNP2!_)
                   DF3(DNP2,DNP0!_)     DF3(DNP1,DNP0!_,DNP2)
                   DF3(DNP2,DNP1!_)     DF3(DNP2,DNP0!_,DNP1!_)
```

Translating identically named sites

Coming back to the immune response rules

1. Rule(s) governing free DF3 binding IgE

BNGL:

```
IgE(Fab)+DF3(DNP,DNP,DNP) -> IgE(Fab!1).DF3(DNP!1,DNP,DNP) k1
```

Kappa:

```
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0!1,DNP1,DNP2) @ k1
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0,DNP1!1,DNP2) @ k1
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0,DNP1,DNP2!1) @ k1
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab1!1),DF3(DNP0!1,DNP1,DNP2) @ k1
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab1!1),DF3(DNP0,DNP1!1,DNP2) @ k1
IgE(Fab0),DF3(DNP0,DNP1,DNP2) -> IgE(Fab1!1),DF3(DNP0,DNP1,DNP2!1) @ k1
```

Translating identically named sites

2. Rule(s) governing IgE crosslinking by DF3

BNGL:

```
IgE(Fab)+DF3(DNP,DNP!+) -> IgE(Fab!1).DF3(DNP!1,DNP!+) k2
```

Kappa:

```
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0!1,DNP1!_,DNP2) @ k2
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0,DNP1!_,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!1,DNP1!_,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!1,DNP1,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0,DNP1!1,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!1,DNP2) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!_,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!1,DNP2!_) @ k2
```

Translating identically named sites

2. Rule(s) governing IgE crosslinking by DF3

BNGL:

```
IgE(Fab)+DF3(DNP,DNP!+) -> IgE(Fab!1).DF3(DNP!1,DNP!+) k2
```

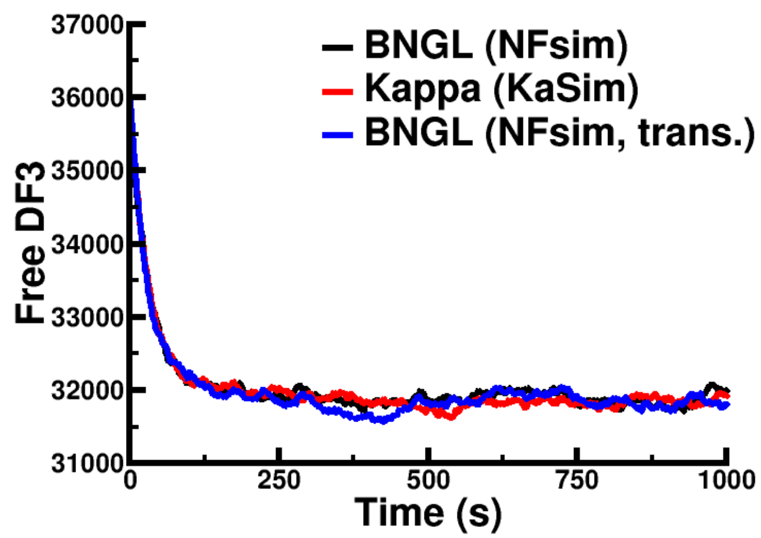
Kappa:

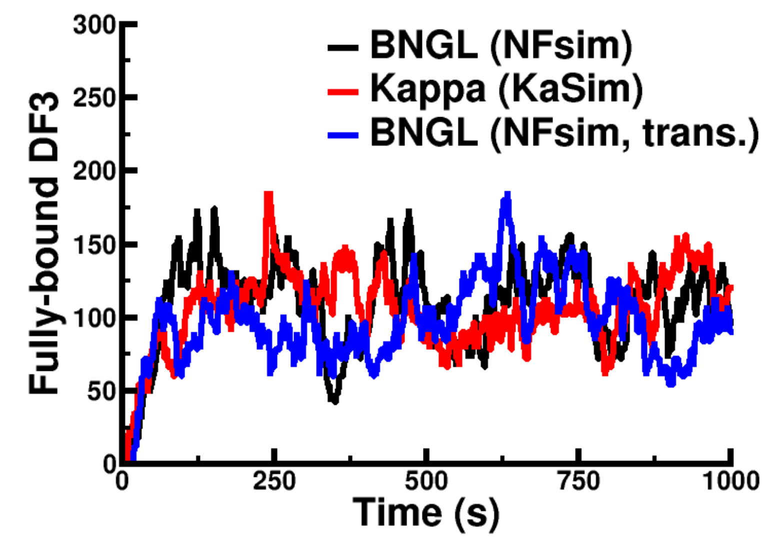
```
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0!1,DNP1!_,DNP2) @ k2
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0,DNP1!_,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0,DNP1!_,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!1,DNP1!_,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!1,DNP1,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0,DNP1!1,DNP2!_) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!1,DNP2) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1!_,DNP2) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!_,DNP2!1) @ k2
IgE(Fab0),DF3(DNP0!_,DNP1,DNP2!_) -> IgE(Fab0!1),DF3(DNP0!_,DNP1!1,DNP2!_) @ k2
```

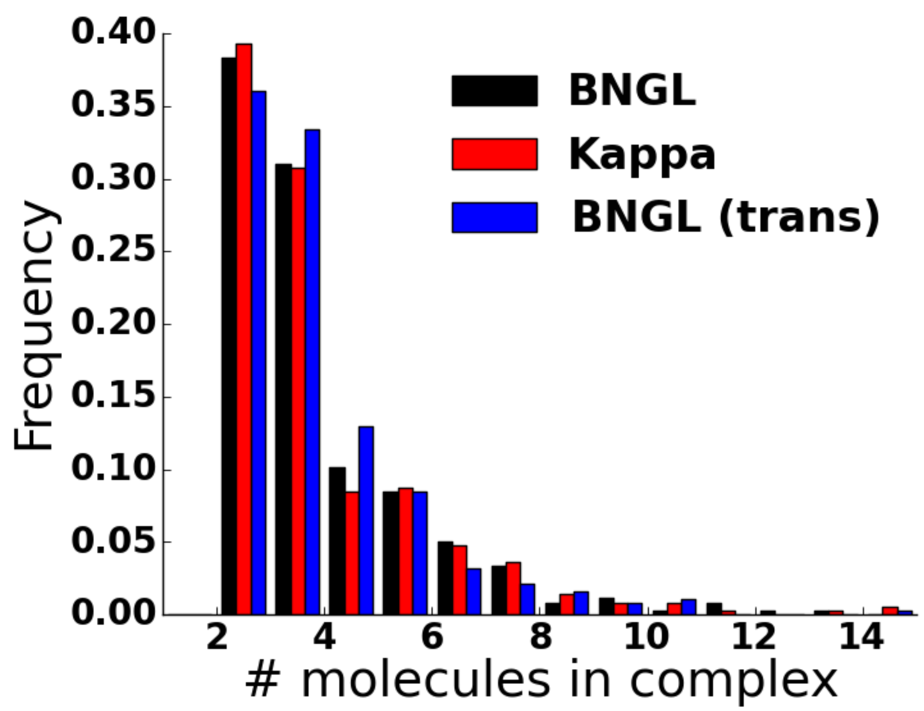
Simulation results

The complete model also includes fully independent unbinding

BNGL to Kappa and back to BNGL translations result in identical simulation trajectories







All models also capture similar aggregate size distributions

Los Alamos
NATIONAL LABORATORY
EST. 1943

Center for
Nonlinear Studies

Additional considerations

Include Kappa tokens and BNGL populations

Integrate SBML **multi** extension, other languages

Acknowledgments

People:

Bill Hlavacek

Song Feng

Yen Ting Lin

Eshan Mitra

Alex Ionkov

Funding:

NIH NIGMS

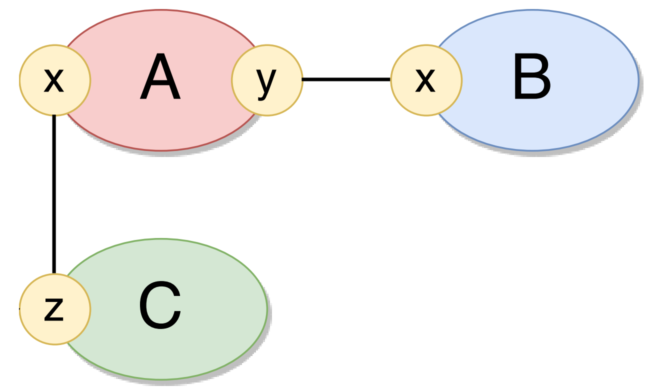
Center for Nonlinear Studies

Los Alamos National Laboratory EST. 1943

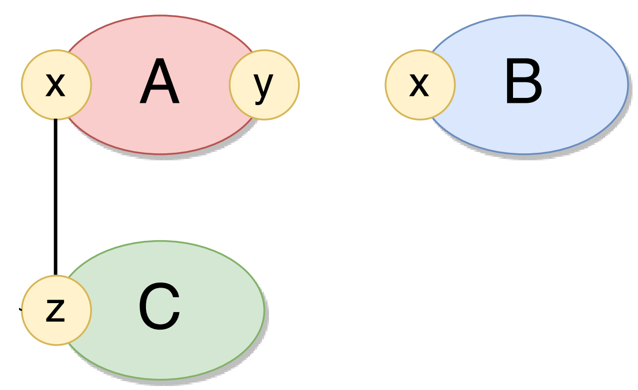
Molecularity

BNGL operators enforce molecularity on pattern matching

```
A(x!+).B() =>

A(x!_),B() =>
```




```
A(x!+).B() =/>

A(x!+)+B() =>

A(x!_),B() =>
```



Kappa rules do not (locality)

Grammars

BNGL simple patterns

$$\langle \mathit{pattern} \rangle ::= \text{`}\texttt{0}\text{'} \mid \langle \mathit{molecule} \rangle, [\{\text{`}\texttt{.}\text{'}, \langle \mathit{molecule} \rangle\}]$$

$$\langle \mathit{molecule} \rangle ::= \langle \mathit{bName} \rangle, [\text{`}\texttt{(}\text{'}, \langle \mathit{compList} \rangle\ \text{`}\texttt{)}\text{'}]$$

$$\langle \mathit{compList} \rangle ::= \langle \mathit{empty} \rangle \mid \langle \mathit{component} \rangle, [\{\text{`}\texttt{,}\text{'}, \langle \mathit{component} \rangle\}]$$

$$\langle \mathit{component} \rangle ::= \langle \mathit{bName} \rangle, \langle \mathit{compState} \rangle, \langle \mathit{compBond} \rangle$$

$$\langle \mathit{compState} \rangle ::= \langle \mathit{empty} \rangle \mid \text{`}\texttt{\textasciitilde}\text{'}, \langle \mathit{bName} \rangle$$

$$\langle \mathit{compBond} \rangle ::= \langle \mathit{empty} \rangle \mid \text{`}\texttt{!?}\text{'} \mid \text{`}\texttt{!+}\text{'} \mid \text{`}\texttt{!}\text{'}, \langle \mathit{integer} \rangle$$

Kappa simple patterns

$$\langle \mathit{pattern} \rangle ::= \langle \mathit{empty} \rangle \mid \langle \mathit{agent} \rangle, [\{\text{`}\texttt{,}\text{'}, \langle \mathit{agent} \rangle\}]$$

$$\langle \mathit{agent} \rangle ::= \langle \mathit{kName} \rangle, \text{`}\texttt{(}\text{'}, \langle \mathit{siteList} \rangle, \text{`}\texttt{)}\text{'}$$

$$\langle \mathit{siteList} \rangle ::= \langle \mathit{empty} \rangle \mid \langle \mathit{site} \rangle, [\{\text{`}\texttt{,}\text{'}, \langle \mathit{site} \rangle\}]$$

$$\langle \mathit{site} \rangle ::= \langle \mathit{kName} \rangle, \langle \mathit{siteState} \rangle, \langle \mathit{bond} \rangle$$

$$\langle \mathit{siteState} \rangle ::= \langle \mathit{empty} \rangle \mid \text{`}\texttt{\textasciitilde}\text{'}, \langle \mathit{kName} \rangle$$

$$\langle \mathit{bond} \rangle ::= \langle \mathit{empty} \rangle \mid \text{`}\texttt{!}\text{'}, \langle \mathit{integer} \rangle \mid \text{`}\texttt{!\_}\text{'} \mid \text{`}\texttt{?}\text{'}$$

Useful regular expressions

$$\begin{aligned}
\langle \mathit{integer} \rangle &= [\text{0-9}]+ \\
\langle \mathit{bName} \rangle &= [\text{a-zA-Z}][\text{a-zA-Z_0-9}]* \\
\langle \mathit{kName} \rangle &= [\text{a-zA-Z}][\text{a-zA-Z_0-9+-}]* \\
\langle \mathit{string} \rangle &= .*
\end{aligned}$$

Grammars

BNGL simple rules

$$\langle \mathit{uniRule} \rangle ::= [\langle \mathit{bName} \rangle,\ \text{‘:’}],\ \langle \mathit{patternList} \rangle,\ \text{‘->’},\ \langle \mathit{patternList} \rangle,\ \langle \mathit{ws} \rangle,\ \langle \mathit{rate} \rangle,\ \langle \mathit{newline} \rangle$$

$$\langle \mathit{biRule} \rangle ::= [\langle \mathit{bName} \rangle,\ \text{‘:’}],\ \langle \mathit{patternList} \rangle,\ \text{‘<->’},\ \langle \mathit{patternList} \rangle,\ \langle \mathit{ws} \rangle,\ \langle \mathit{rate} \rangle,\ \langle \mathit{rate} \rangle,\ \langle \mathit{newline} \rangle$$

$$\langle \mathit{patternList} \rangle ::= \langle \mathit{empty} \rangle \mid \langle \mathit{pattern} \rangle,\ \text{‘+’},\ \langle \mathit{patternList} \rangle$$

$$\langle \mathit{rate} \rangle ::= \text{? an algebraic expression in BNGL syntax ?}$$

Kappa simple rules

$$\langle \mathit{uniRule} \rangle ::= [\text{‘‘’},\ \langle \mathit{string} \rangle,\ \text{‘’’}],\ \langle \mathit{pattern} \rangle,\ \text{‘->’},\ \langle \mathit{pattern} \rangle,\ \text{‘@’},\ \langle \mathit{rate} \rangle,\ \langle \mathit{newline} \rangle$$

$$\langle \mathit{biRule} \rangle ::= [\text{‘‘’},\ \langle \mathit{string} \rangle,\ \text{‘’’}],\ \langle \mathit{pattern} \rangle,\ \text{‘<->’},\ \langle \mathit{pattern} \rangle,\ \text{‘@’},\ \langle \mathit{rate} \rangle,\ \langle \mathit{rate} \rangle,\ \langle \mathit{newline} \rangle$$

$$\langle \mathit{rate} \rangle ::= \langle \mathit{expression} \rangle,\ [\text{‘\{’},\ \langle \mathit{expression} \rangle\ \text{’\}’}]$$

$$\langle \mathit{expression} \rangle ::= \text{? an algebraic expression in Kappa syntax ?}$$

Part of a model of pheromone signaling in baker's yeast

Readable?

Extensible?

ODE functions:

$$\frac{d[Ste2]}{dt} = -k1[\alpha - factor][Ste2] + k2[Ste2_{active}] - k7[Ste2] + \frac{k4[Ste12_{active}]^2}{k5^2 + [Ste12_{active}]^2} + k6$$

$$\frac{d[Ste2_{active}]}{dt} = k1[\alpha - factor][Ste2] - k2[Ste2_{active}] - k3[Ste2_{active}]$$

$$\frac{d[Sst2_{active}]}{dt} = \frac{k44[Ste12_{active}]^2}{k45^2 + [Ste12_{active}]^2} - k46[Sst2_{active}]$$

$$\frac{d[G]}{dt} = -k8[Ste2_{active}][G] + k15[G_\alpha d][G_\beta\gamma] + \frac{k9[Ste12_{active}]^2}{k10^2 + [Ste12_{active}]^2} - k12[G] + k11$$

$$\frac{d[G\alpha t]}{dt} = k8[Ste2_{active}][G] - k13[G\alpha t] - k14[G\alpha t][Sst2_{active}]$$

$$\frac{d[G_\alpha d]}{dt} = k13[G\alpha t] + k14[G\alpha t][Sst2_{active}] - k15[G_\alpha d][G_\beta\gamma]$$

$$\frac{d[G_\beta\gamma]}{dt} = k8[Ste2_{active}][G] - k15[G_\alpha d][G_\beta\gamma] - k40[G_\beta\gamma][Far1pp_{out}] + k41[Far1pp_{out}G_\beta\gamma]20 - k18[G_\beta\gamma][Ste20] + k19[G_\beta\gamma Ste20]$$

$$\frac{d[Ste20]}{dt} = -k18[G_\beta\gamma][Ste20] + k19[G_\beta\gamma Ste20]$$

$$\begin{aligned}\frac{d[G_\beta\gamma Ste20]}{dt} = {} & k18[G_\beta\gamma][Ste20] - k19[G_\beta\gamma Ste20] - k16[G_\beta\gamma Ste20]B1 + k17C1 - k16[G_\beta\gamma Ste20]B2 + k17C2 - k16[G_\beta\gamma Ste20]B3 + k17C3 - \\ & k16[G_\beta\gamma Ste20]B4 + k17C4 - k16[G_\beta\gamma Ste20]B5 + k17C5 - k16[G_\beta\gamma Ste20]B6 + k17C6 - k16[G_\beta\gamma Ste20]B7 + k17C7 - k16[G_\beta\gamma Ste20]B8 + \\ & k17C8 - k16[G_\beta\gamma Ste20]B9 + k17C9 - k16[G_\beta\gamma Ste20]B10 + k17C10 - k16[G_\beta\gamma Ste20]B11 + k17C11 - k16[G_\beta\gamma Ste20]B12 + k17C12 - \\ & k16[G_\beta\gamma Ste20]B13 + k17C13 - k16[G_\beta\gamma Ste20]B14 + k17C14 - k16[G_\beta\gamma Ste20]B15 + k17C15 - k16[G_\beta\gamma Ste20]B16 + k17C16 - \\ & k16[G_\beta\gamma Ste20]B17 + k17C17 - k16[G_\beta\gamma Ste20]B18 + k17C18 - k16[G_\beta\gamma Ste20]B19 + k17C19 - k16[G_\beta\gamma Ste20]B20 + k17C20 - \\ & k16[G_\beta\gamma Ste20]B21 + k17C21 - k16[G_\beta\gamma Ste20]B22 + k17C22 - k16[G_\beta\gamma Ste20]B23 + k17C23 - k16[G_\beta\gamma Ste20]B24 + k17C24 - \\ & k16[G_\beta\gamma Ste20]B25 + k17C25 - k16[G_\beta\gamma Ste20]B26 + k17C26 - k16[G_\beta\gamma Ste20]B27 + k17C27\end{aligned}$$

$$\begin{aligned}\frac{d[Ste11]}{dt} = {} & p1_{KKK}[Ste11pMAPKKK-P] + off_{KKK}(C2+C8+C11+C12+C15+C20+C22+C23+C26+B2+B8+B11+B12+B15+B20+B22+ \\ & B23+B26) - on_{KKK}[Ste11](C4+C6+C7+C1+C16+C17+C18+C19+C5+B4+B6+B7+B1+B16+B17+B18+B19+B5) - k26[Ste11][Fus3pp_{out}]\end{aligned}$$

$$\begin{aligned}\frac{d[Ste11p]}{dt} = {} & -a1_{KKK}[Ste11p]([MAPKKK-P]_0 - [Ste11pMAPKKK-P] - [Ste11ppMAPKKK-P]) + d1_{KKK}[Ste11pMAPKKK-P] + \\ & p2_{KKK}[Ste11ppMAPKKK-P]\end{aligned}$$

$$\begin{aligned}\frac{d[Ste11pMAPKKK-P]}{dt} = {} & \\ & a1_{KKK}[Ste11p]([MAPKKK-P]_0 - [Ste11pMAPKKK-P] - [Ste11ppMAPKKK-P]) - (d1_{KKK} + p1_{KKK})[Ste11pMAPKKK-P]\end{aligned}$$

$$\begin{aligned}\frac{d[Ste11pp]}{dt} = {} & -a2_{KKK}[Ste11pp]([MAPKKK-P]_0 - [Ste11pMAPKKK-P] - [Ste11ppMAPKKK-P]) + d2_{KKK}[Ste11ppMAPKKK - \\ & P] - a3_{KK}[Ste11pp][Ste7] + (d3_{KK} + p3_{KK})[Ste11ppSte7] - a4_{KK}[Ste11pp][Ste7p] + (d4_{KK} + p4_{KK})[Ste11ppSte7p] + **off_{KKK}(C3 + C10 + \\ & C9 + C13 + C14 + C21 + C24 + C25 + C27 + B3 + B10 + B9 + B13 + B14 + B21 + B24 + B25 + B27) - **on_{KKK}[Ste11pp](C1 + C4 + C5 + \\ & C6 + C7 + C16 + C17 + C18 + C19 + B1 + B4 + B5 + B6 + B7 + B16 + B17 + B18 + B19)\end{aligned}$$

$$\begin{aligned}\frac{d[Ste11ppMAPKKK-P]}{dt} = {} & \\ & a2_{KKK}[Ste11pp]([MAPKKK-P]_0 - [Ste11pMAPKKK-P] - [Ste11ppMAPKKK-P]) - (d2_{KKK} + p2_{KKK})[Ste11ppMAPKKK-P]\end{aligned}$$

$$\begin{aligned}\frac{d[Ste7]}{dt} = {} & -a3_{KK}[Ste7][Ste11pp] + d3_{KK}[Ste11ppSte7] + p1_{KK}[Ste7pMAPKK-P] + off_{KK}(C4 + C8 + C9 + C16 + C19 + C20 + C21 + \\ & C23 + C25 + B4 + B8 + B9 + B16 + B19 + B20 + B21 + B23 + B25) - on_{KK}[Ste7](C1 + C2 + C3 + C6 + C7 + C12 + C13 + C14 + C15 + B1 + \\ & B2 + B3 + B6 + B7 + B12 + B13 + B14 + B15) - k24[Ste7][Fus3_{out}] + k25[Fus3_{out}Ste7]\end{aligned}$$

$$\frac{d[Ste7Ste11pp]}{dt} = a3_{KK}[Ste11pp][Ste7] - (d3_{KK} + p3_{KK})[Ste11ppSte7]$$

$$\begin{aligned}\frac{d[Ste7p]}{dt} = {} & -a1_{KK}([MAPKK-P]_0 - [Ste7pMAPKK-P] - [Ste7ppMAPKK-P])[Ste7p] + d1_{KK}[Ste7pMAPKK-P] + \\ & p3_{KK}[Ste11ppSte7] - a4_{KK}[Ste7p][Ste11pp] + d4_{KK}[Ste11ppSte7p] + p2_{KK}[Ste7ppMAPKK-P]\end{aligned}$$

$$\frac{d[Ste7pMAPKK-P]}{dt} = a1_{KK}[Ste7p]([MAPKK-P]_0 - [Ste7pMAPKK-P] - [Ste7ppMAPKK-P]) - (d1_{KK} + p1_{KK})[Ste7pMAPKK-P]$$

$$\frac{d[Ste7pSte11pp]}{dt} = a4_{KK}[Ste11pp][Ste7p] - (d4_{KK} + p4_{KK})[Ste11ppSte7p]$$

$$\begin{aligned}\frac{d[Ste7pp]}{dt} = {} & -a2_{KK}[Ste7pp]([MAPKK-P]_0 - [Ste7pMAPKK-P] - [Ste7ppMAPKK-P]) + d2_{KK}[Ste7ppMAPKK-P] + \\ & p4_{KK}[Ste11ppSte7p] - a3_K[Ste7pp][Fus3_{out}] + (d3_K + p3_K)[Ste7ppFus3_{out}] - a4_K[Ste7pp][Fus3p_{out}] + (d4_K + p4_K)[Ste7ppFus3p_{out}] + ** \\ & off_{KK}(C5 + C10 + C11 + C17 + C22 + C24 + B5 + B10 + B11 + B17 + B22 + B24) + **off'_{KK}(C18 + C26 + C27 + B18 + B26 + B27) - k27[Ste7pp]\end{aligned}$$

$$\frac{d[Ste7ppMAPKK-P]}{dt} = a2_{KK}[Ste7pp]([MAPKK-P]_0 - [Ste7pMAPKK-P] - [Ste7ppMAPKK-P]) - (d2_{KK} + p2_{KK})[Ste7ppMAPKK-P]$$

$$\begin{aligned}\frac{d[Fus3_{out}]}{dt} = {} & -a3_K[Ste7pp][Fus3_{out}] + d3_K[Ste7ppFus3_{out}] + p1_K[Fus3p_{out}MAPK-P_{out}] + off_K(C6 + C12 + C13 + C16 + C17 + C20 + \\ & C21 + C22 + C24 + B6 + B12 + B13 + B16 + B17 + B20 + B21 + B22 + B24) - on_K[Fus3_{out}](C1 + C2 + C3 + C4 + C5 + C8 + C9 + C10 + C11 + \\ & B1 + B2 + B3 + B4 + B5 + B8 + B9 + B10 + B11) - k24[Ste7][Fus3_{out}] + k25[Fus3_{out}Ste7] + k47[Fus3_{in}] - k48[Fus3_{out}] + \frac{k32[Ste12_{active}]^2}{k5^2 + [Ste12_{active}]^2}\end{aligned}$$

$$\frac{d[Fus3_{out}Ste11pp]}{dt} = a3_K[Ste7pp][Fus3_{out}] - (d3_K + p3_K)[Ste7ppFus3_{out}]$$

$$\begin{aligned}\frac{d[Fus3p_{out}]}{dt} = {} & -a1_K[Fus3p_{out}][MAPK-P_{out}] + d1_K[Fus3p_{out}MAPK-P_{out}] + p3_K[Ste7ppFus3_{out}] - a4_K[Ste7pp][Fus3p_{out}] + \\ & d4_K[Ste7ppFus3p_{out}] + p2_K[Fus3pp_{out}MAPK-P_{out}]\end{aligned}$$

$$\frac{d[Fus3p_{out}MAPK-P_{out}]}{dt} = a1_K[Fus3p_{out}][MAPK-P_{out}] - (d1_K + p1_K)[Fus3p_{out}MAPK-P_{out}]$$

$$\frac{d[Fus3p_{out}STe7pp]}{dt} = a4_K[Ste7pp][Fus3p_{out}] - (d4_K + p4_K)[Ste7ppFus3p_{out}]$$

$$\begin{aligned}\frac{d[Fus3pp_{out}]}{dt} = {} & -a2_K[Fus3pp_{out}][MAPK-P_{out}] + d2_K[Fus3pp_{out}MAPK-P_{out}] + p4_K[Ste7ppFus3p_{out}] + **off_K(C7 + C14 + C15 + \\ & C18 + C19 + C23 + C25 + C26 + C27 + B7 + B14 + B15 + B18 + B19 + B23 + B25 + B26 + B27) + k49[Fus3pp_{in}] - k50[Fus3pp_{out}]\end{aligned}$$

$$\frac{d[Fus3pp_{out}MAPK-P_{out}]}{dt} = a2_K[Fus3pp_{out}][MAPK-P_{out}] - (d2_K + p2_K)[Fus3pp_{out}MAPK-P_{out}]$$

$$\begin{aligned}\frac{d[MAPK-P_{out}]}{dt} = {} & -a1_K[Fus3p_{out}][MAPK-P_{out}] + (d1_K + p1_K)[Fus3p_{out}MAPK-P_{out}] - a2_K[Fus3pp_{out}][MAPK-P_{out}] + (p2_K + \\ & d2_K)[Fus3pp_{out}MAPK-P_{out}] + \frac{k31[Ste12_{active}]^2}{k5^2 + [Ste12_{active}]^2}\end{aligned}$$

$$\frac{d[Fus3_{out}Ste7]}{dt} = k24[Ste7][Fus3_{out}] - k25[Fus3_{out}Ste7]$$

Shao, D,, et al, (2006), Biophys J

Los Alamos NATIONAL LABORATORY EST.1943

Center for Nonlinear Studies

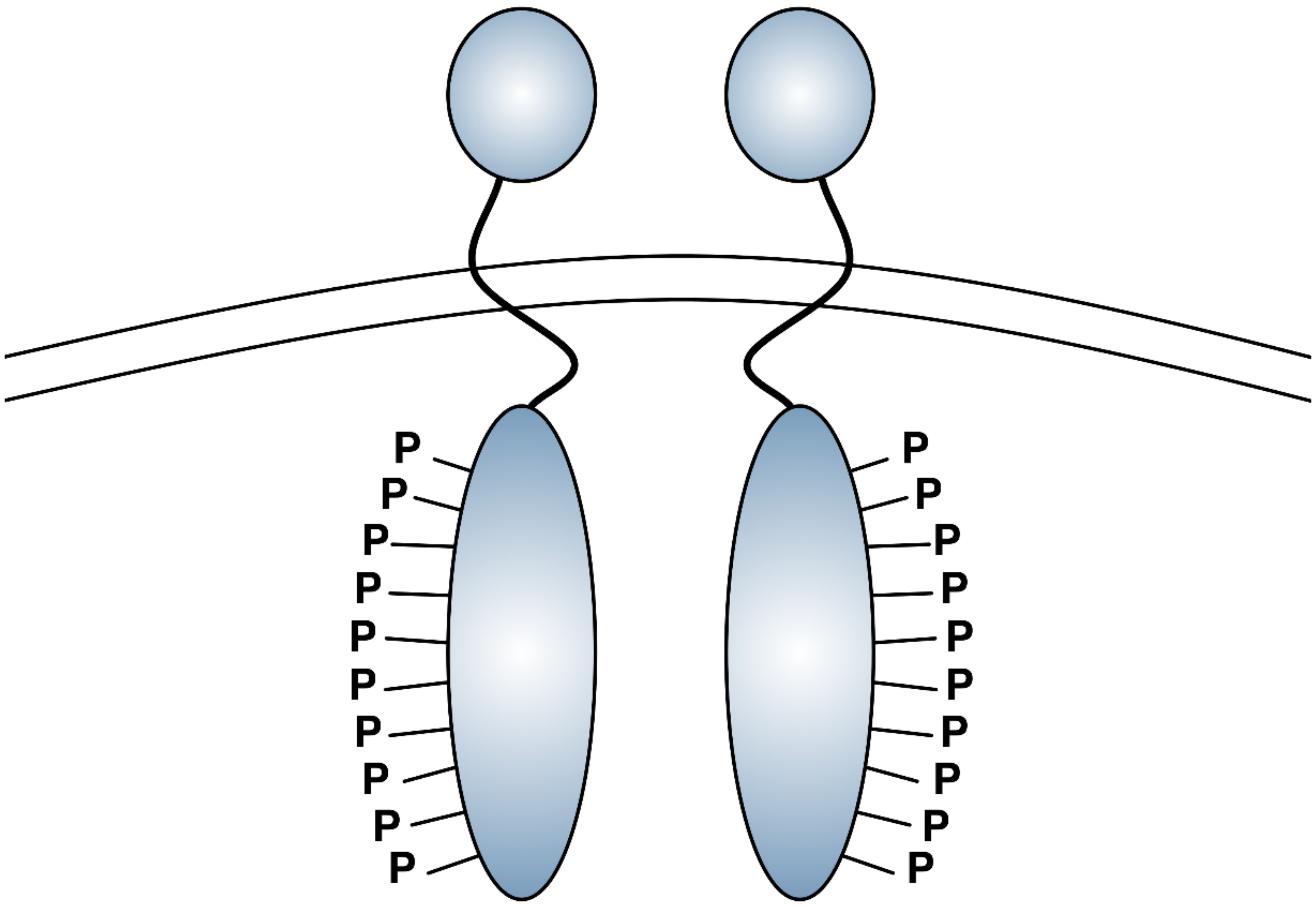
Dynamical systems biology

Modeling protein interaction networks traditionally done with ODEs or reaction networks

Two prominent issues:

Encoding (knowledge representation)

Combinatorial complexity

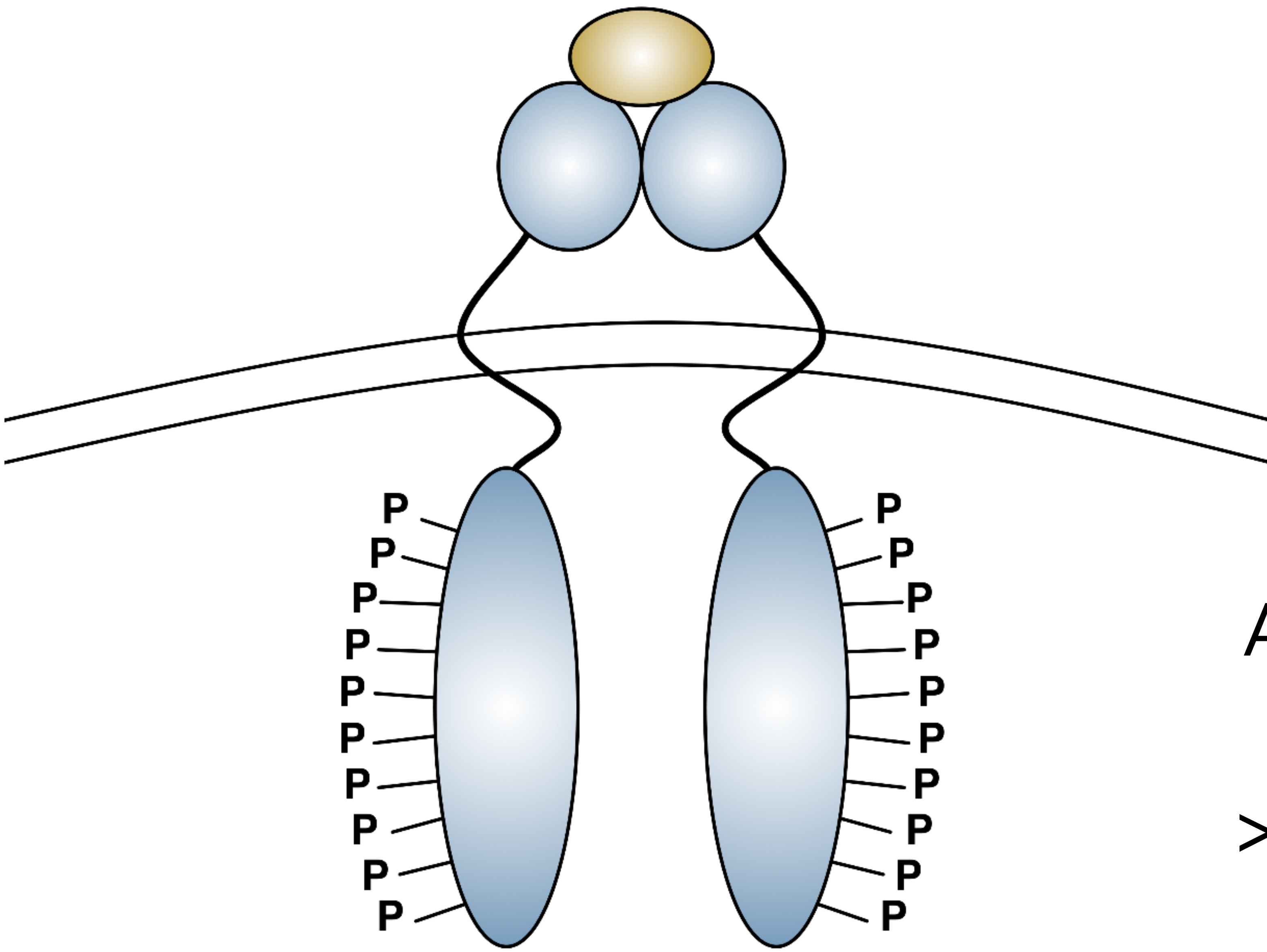


PDGF receptor

10 phosphorylation sites

2^{10} possible states

Combinatorial complexity



PDGF receptor

10 phosphorylation sites

2^{10} possible states

Active receptor dimerizes

>500,000 possible states

Los Alamos NATIONAL LABORATORY EST. 1943

Center for Nonlinear Studies