Supplementary Information

A. Sample BioNetGen input file for simulating the early response of RBL cells stimulated by a reversible bivalent ligand that binds to, and dimerizes, IgE-Fc complexes on RBL cell surfaces. In this input file, the first and second ligand dissociation rate constants k_{-1} and k_{-2} are given by $k_{-1}=k_{-2}= 10s^{-1}$. The total number of receptors per RBL cell, $N_T = 4*10^5$ molecules and the surface area of the cell is taken as $A_{cell} = 8 * 10^{-6}$ cm^2, so that the total receptor surface density is given by $R_T = 5*10^{10}$ molecules cm^{-2} . The first ligand-receptor binding rate constant $k_{+1}= 10^6$ $M^{-1}s^{-1}$. This second order rate constant is divided by the inverse RBL cell density in order to obtain the rate constant in units of $molecule^{-1}s^{-1}$ used in the BioNetGen input file. Assuming RBL cell density to be of 10^6 cells/ml [1], i.e. an inverse cell density (V_{cell}) of 10^{-6} ml/cell, k_{+1} reduces to $1.66058*10^{-9}$ $molecule^{-1}$ s^{-1}. The cross-linking rate constant k_{+2} is chosen as $2*10^{-11}$ $molecule^{-1}$ cm^2 s^{-1} so that $k_{+2}R_T = 1.0s^{-1}$. This rate constant is divided by the RBL surface area A_{cell}, to obtain the rate constant in units of $molecule^{-1}$ s^{-1} used in the BioNetGen input file. All the other rate constants and total cellular Lyn and Syk expression levels are taken from Faeder et al . JI 2003. The total number of ligands per RBL cell, L_T, is chosen in such a way that $2K_1L =1$, where L is the free ligand concentration at equilibrium and $K_1 = k_{+1}/k_{-1}$.

```
begin parameters
Lig_tot        3.0112e9
Rec_tot        4.0e5
Lyn_tot        2.8e4
Syk_tot        4.0e5

kp1  1.66058e-9
km1  10.0
kp2  2.5e-6
km2  10.0
kpL  5e-5
kmL  20
kpLs  5e-5
kmLs  0.12
kpS  6e-5
kmS  0.13
kpSs  6e-5
kmSs  0.13
pLb  30
pLbs  100
pLg   1
pLgs   3
pLS   30
pLSs  100
pSS  100
pSSs  200
dm   20
dc   20
end parameters
```

```
begin species
Lig(l,l) Lig_tot
Lyn(U,SH2) Lyn_tot
Syk(tSH2,l~Y,a~Y) Syk_tot
Rec(a,b~Y,g~Y) Rec_tot
end species

begin reaction_rules
# Ligand-receptor binding
1 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1

# Receptor-aggregation
2 Rec(a) + Lig(l,l!1) <-> Rec(a!2).Lig(l!2,l!1) kp2,km2

# Constitutive Lyn-receptor binding
3 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL

# Transphosphorylation of beta by constitutive Lyn
4 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~Y) -> \
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~pY) pLb

# Transphosphorylation of gamma by constitutive Lyn
5 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~Y) -> \
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY) pLg

# Lyn-receptor binding through SH2 domain
6 Rec(b~pY) + Lyn(U,SH2) <-> Rec(b~pY!1).Lyn(U,SH2!1) kpLs, kmLs

# Transphosphorylation of beta by SH2-bound Lyn
7 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~Y) -> \
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~pY) pLbs

# Transphosphorylation of gamma by SH2-bound Lyn
8 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~Y) -> \
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY) pLgs

# Syk-receptor binding through tSH2 domain
9 Rec(g~pY) + Syk(tSH2) <-> Rec(g~pY!1).Syk(tSH2!1) kpS, kmS

# Transphosphorylation of Syk by constitutive Lyn
10 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \
   Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLS

# Transphosphorylation of Syk by SH2-bound Lyn
11 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \
   Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLSs

# Transphosphorylation of Syk by Syk not phosphorylated on aloop
12 Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \
   Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSS
```

```
# Transphosphorylation of Syk by Syk phosphorylated on aloop
13 Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \
  Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSSs

# Dephosphorylation of Rec beta
14 Rec(b~pY)-> Rec(b~Y) dm

# Dephosphorylation of Rec gamma
15 Rec(g~pY)-> Rec(g~Y) dm

# Dephosphorylation of Syk at membrane
16 Syk(tSH2!1,l~pY)-> Syk(tSH2!1,l~Y) dm
17 Syk(tSH2!1,a~pY)-> Syk(tSH2!1,a~Y) dm

# Dephosphorylation of Syk in cytosol
18 Syk(tSH2,l~pY)-> Syk(tSH2,l~Y) dc
19 Syk(tSH2,a~pY)-> Syk(tSH2,a~Y) dc

end reaction_rules

begin observables
Species       Ligfree     Lig(l,l)
Species       RecMon      Rec(a) Rec(a!1).Lig(l!1,l)
Species       RecDim      Rec(a!1).Lig(l!1,l!2).Rec(a!2)
Molecules     RecPbeta    Rec(b~pY!?)
Molecules     RecPgamma   Rec(g~pY!?)
Molecules     RecSykPS    Rec(g~pY!1).Syk(tSH2!1,a~pY)
end observables

generate_network({overwrite=>1});
simulate_ode({suffix=>"3.01120e+09",t_end=>1000000,n_steps=>50000, \
atol=>1e-11,rtol=>1e-11,steady_state=>1});
```

B. Sample BioNetGen input file for simulating the early response of RBL cells stimulated by a reversible trivalent ligand that binds to, and forms trimeric IgE-Fc complexes on RBL cell surfaces. For the trivalent ligand, it is assumed that $k_{-1}=k_{-2}=k_{-3}$ and the k_{-1} and k_{-2} values are the same as in the bivalent ligand case and $k_{+3}=k_{+2}$ where k_{+2} is the same as in the bivalent ligand case. All other rate constants and total cellular concentrations are taken to be the same as in the bivalent ligand case.

```
begin parameters
Lig_tot       3.0112e9
Rec_tot       4.0e5
Lyn_tot       2.8e4
Syk_tot       4.0e5

kp1  1.66058e-9
km1  10.0
kp2  2.5e-6
```

```
km2  10.0
kpL  5e-5
kmL  20
kpLs  5e-5
kmLs  0.12
kpS  6e-5
kmS  0.13
kpSs  6e-5
kmSs  0.13
pLb  30
pLbs  100
pLg   1
pLgs   3
pLS   30
pLSs  100
pSS  100
pSSs  200
dm   20
dc   20
end parameters

begin species
Lig(l,l,l) Lig_tot
Lyn(U,SH2) Lyn_tot
Syk(tSH2,l~Y,a~Y) Syk_tot
Rec(a,b~Y,g~Y) Rec_tot
end species

begin reaction_rules
# Free ligand - receptor binding
1 Rec(a) + Lig(l,l,l) <-> Rec(a!1).Lig(l!1,l,l) kp1, km1

# Receptor-aggregation (ligand is already bound at one site)
2 Rec(a) + Lig(l,l,l!+) <-> Rec(a!1).Lig(l!1,l,l!+) kp2, km2

# Receptor-aggregation (ligand is already bound at two sites)
3 Rec(a) + Lig(l,l!+,l!+) <-> Rec(a!1).Lig(l!1,l!+,l!+) kp2, km2

# Constitutive Lyn-receptor binding
4 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL

# Transphosphorylation of beta by constitutive Lyn
5 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~Y) -> \
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~pY) pLb

# Transphosphorylation of gamma by constitutive Lyn
6 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~Y) -> \
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY) pLg

# Lyn-receptor binding through SH2 domain
7 Rec(b~pY) + Lyn(U,SH2) <-> Rec(b~pY!1).Lyn(U,SH2!1) kpLs, kmLs
```

```
# Transphosphorylation of beta by SH2-bound Lyn
8 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~Y) -> \
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~pY) pLbs

# Transphosphorylation of gamma by SH2-bound Lyn
9 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~Y) -> \
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY) pLgs

# Syk-receptor binding through tSH2 domain
10 Rec(g~pY) + Syk(tSH2) <-> Rec(g~pY!1).Syk(tSH2!1) kpS, kmS

# Transphosphorylation of Syk by constitutive Lyn
11 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \
  Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLS

# Transphosphorylation of Syk by SH2-bound Lyn
12 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \
  Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLSs

# Transphosphorylation of Syk by Syk not phosphorylated on aloop
13 Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \
  Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSS

# Transphosphorylation of Syk by Syk phosphorylated on aloop
14 Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \
  Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSSs

# Dephosphorylation of Rec beta
15 Rec(b~pY)-> Rec(b~Y) dm

# Dephosphorylation of Rec gamma
16 Rec(g~pY)-> Rec(g~Y) dm

# Dephosphorylation of Syk at membrane
17 Syk(tSH2!1,l~pY)-> Syk(tSH2!1,l~Y) dm
18 Syk(tSH2!1,a~pY)-> Syk(tSH2!1,a~Y) dm

# Dephosphorylation of Syk in cytosol
19 Syk(tSH2,l~pY)-> Syk(tSH2,l~Y) dc
20 Syk(tSH2,a~pY)-> Syk(tSH2,a~Y) dc

end reaction_rules

begin observables
Species       Ligfree     Lig(l,l,l)
Species       RecMon      Rec(a) Rec(a!1).Lig(l!1,l,l)
Species       RecDim      Rec(a!1).Lig(l!1,l!2,l).Rec(a!2)
Species       RecTri      Rec(a!1).Lig(l!1,l!2,l!3).Rec(a!2).Rec(a!3)
Molecules     RecPbeta    Rec(b~pY!?)
Molecules     RecPgamma   Rec(g~pY!?)
```

```
Molecules     RecSykPS     Rec(g~pY!1).Syk(tSH2!1,a~pY)
end observables

generate_network({overwrite=>1});
simulate_ode({suffix=>"3.01120e+09",t_end=>1000000,n_steps=>50000, \
atol=>1e-11,rtol=>1e-11,steady_state=>1});
```

C. Sample Perl script to write out a BioNetGen Input file for simulating the early response of RBL cells stimulated by a reversible bivalent ligand that binds to, and dimerizes, IgE-Fc complexes on RBL cell surfaces, and run simulations using this input file for a list of total number of ligands per RBL cell. In this particular example, the total number of ligands per RBL cell, L_T is chosen in such a way that $\log_{10}(2K_1L)$, varies from -10 to 10, where L is the free ligand concentration at equilibrium and $K_1 = k_{+1}/k_{-1}$. Running this Perl script requires the definition of an environment variable $BNGPATH in the script, which defines the path to the directory where the BioNetGen program is installed. This script requires on the command line input one argument which is the name of the directory created on running the Perl script and which contains the BioNetGen input and output files generated. For example, if the lone command line argument is "test" then a directory called "test" is created in the same directory that contains the Perl script. The directory called "test" contains the BioNetGen input file created, "test.bngl", a log file called "test.log" and a network generation output file called "test.net" which corresponds to the total ligand concentration in the parameter block contained in the Perl script. For each L_T value, the "test" directory contains a network file containing the results of network generation in the form of "test_X.XXXXXe±XX.net", and output files containing time course simulation results, "test_X.XXXXXe±XX.cdat" (containing simulated time courses of all species in the model) and "test_X.XXXXXe±XX.gdat" (containing simulated time courses of the species specified in the observable block of the Perl script) where X.XXXXXe±XX is the value of L_T in exponential format.

```
#!/usr/bin/perl

$BNGPATH="/Users/ambarish/bionetgen/BioNetGen_2.0.35" ;
@lig_conc=(0.284552,2.99445,30.1157,301.141,3011.42,30114.2,301142.0,3.01142e6,
3.01142e7,3.01138e8,3.0112e9,3.01104e10,3.011e11,3.011e12,3.011e13,3.011e14,
3.011e15,3.011e16,3.011e17,3.011e18,3.011e19);
($#ARGV==0) || die "Usage $0 runname";
$prefix=shift;
if (-d $prefix){
  die "Directory $prefix exists. Remove before running";
}
$script=<<"EOF";
begin parameters
Lig_tot        3.0112e9
Rec_tot        4.0e5
Lyn_tot        2.8e4
Syk_tot        4.0e5
```

```
kp1  1.66058e-9
km1  10.0
kp2  2.5e-6
km2  10.0
kpL  5e-5
kmL  20
kpLs  5e-5
kmLs  0.12
kpS  6e-5
kmS  0.13
kpSs  6e-5
kmSs  0.13
pLb  30
pLbs  100
pLg   1
pLgs   3
pLS   30
pLSs  100
pSS  100
pSSs  200
dm   20
dc   20
end parameters

begin species
Lig(l,l) Lig_tot
Lyn(U,SH2) Lyn_tot
Syk(tSH2,l~Y,a~Y) Syk_tot
Rec(a,b~Y,g~Y) Rec_tot
end species

begin reaction_rules
# Ligand-receptor binding
1 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1

# Receptor-aggregation
2 Rec(a) + Lig(l,l!1) <-> Rec(a!2).Lig(l!2,l!1) kp2,km2


# Constitutive Lyn-receptor binding
3 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL

# Transphosphorylation of beta by constitutive Lyn
4 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~Y) -> \\
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~pY) pLb

# Transphosphorylation of gamma by constitutive Lyn
5 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~Y) -> \\
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY) pLg

# Lyn-receptor binding through SH2 domain
```

```
6 Rec(b~pY) + Lyn(U,SH2) <-> Rec(b~pY!1).Lyn(U,SH2!1) kpLs, kmLs

# Transphosphorylation of beta by SH2-bound Lyn
7 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~Y) -> \\
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~pY) pLbs

# Transphosphorylation of gamma by SH2-bound Lyn
8 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~Y) -> \\
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY) pLgs

# Syk-receptor binding through tSH2 domain
9 Rec(g~pY) + Syk(tSH2) <-> Rec(g~pY!1).Syk(tSH2!1) kpS, kmS

# Transphosphorylation of Syk by constitutive Lyn
10 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \\
   Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLS

# Transphosphorylation of Syk by SH2-bound Lyn
11 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \\
   Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLSs

# Transphosphorylation of Syk by Syk not phosphorylated on aloop
12 Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \\
   Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSS

# Transphosphorylation of Syk by Syk phosphorylated on aloop
13 Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \\
   Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSSs

# Dephosphorylation of Rec beta
14 Rec(b~pY)-> Rec(b~Y) dm

# Dephosphorylation of Rec gamma
15 Rec(g~pY)-> Rec(g~Y) dm

# Dephosphorylation of Syk at membrane
16 Syk(tSH2!1,l~pY)-> Syk(tSH2!1,l~Y) dm
17 Syk(tSH2!1,a~pY)-> Syk(tSH2!1,a~Y) dm

# Dephosphorylation of Syk in cytosol
18 Syk(tSH2,l~pY)-> Syk(tSH2,l~Y) dc
19 Syk(tSH2,a~pY)-> Syk(tSH2,a~Y) dc

end reaction_rules

begin observables
Species       Ligfree     Lig(l,l)
Species       RecMon      Rec(a) Rec(a!1).Lig(l!1,l)
Species       RecDim      Rec(a!1).Lig(l!1,l!2).Rec(a!2)
Molecules     RecPbeta    Rec(b~pY!?)
Molecules     RecPgamma   Rec(g~pY!?)
```

```
Molecules     RecSykPS    Rec(g~pY!1).Syk(tSH2!1,a~pY)
end observables

generate_network({overwrite=>1});
EOF

mkdir $prefix ;
chdir $prefix;

$fname = sprintf "${prefix}.bngl";
open(BNGL,">$fname") || die "Couldn't write to $fname" ;
print BNGL $script ;
$irun = 0 ;
foreach $i(@lig_conc){
  $irun += 1;
  $srun = sprintf "%6.5e",$i;
  if ($irun > 1){
    print BNGL "resetConcentrations()\n";
  }
  printf BNGL "setParameter(Lig_tot,$i);\n";
  printf BNGL "simulate_ode({suffix=>\"$srun\",t_end=>1000000,n_steps=>50000, \\
  atol=>1e-11,rtol=>1e-11,steady_state=>1});\n";
}
close(BNGL);

# Run BioNetGen on file
print "Running BioNetGen on $fname\n";
system("${BNGPATH}/Perl2/BNG2.pl $fname > $prefix.log");
```

D. Sample Perl script to write out a BioNetGen Input file for simulating the early response of RBL cells stimulated by a reversible trivalent ligand that binds to, and trimerizes, IgE-Fc complexes on RBL cell surfaces, and run simulations using this input file for a list of total number of ligands per RBL cell. For the trivalent ligand, it is assumed that k-1=k-2=k-3 and the k-1 and k-2 values are the same as in the bivalent ligand case and k+3=k+2 where k+2 is the same as in the bivalent ligand case. All other rate constants and total cellular concentrations are taken to be the same as in the bivalent ligand case.

```
#!/usr/bin/perl

$BNGPATH="/Users/ambarish/bionetgen/BioNetGen_2.0.35" ;
@lig_conc=(0.284552,2.99445,30.1157,301.141,3011.42,30114.2,301142.0,3.01142e6,
3.01142e7,3.01138e8,3.0112e9,3.01104e10,3.011e11,3.011e12,3.011e13,3.011e14,
3.011e15,3.011e16,3.011e17,3.011e18,3.011e19);
($#ARGV==0) || die "Usage $0 runname";
$prefix=shift;
if (-d $prefix){
  die "Directory $prefix exists. Remove before running";
}
$script=<<"EOF";
begin parameters
Lig_tot         3.0112e9
```

```
Rec_tot        4.0e5
Lyn_tot        2.8e4
Syk_tot        4.0e5

kp1  1.66058e-9
km1  10.0
kp2  2.5e-6
km2  10.0
kpL  5e-5
kmL  20
kpLs  5e-5
kmLs  0.12
kpS  6e-5
kmS  0.13
kpSs  6e-5
kmSs  0.13
pLb  30
pLbs  100
pLg   1
pLgs   3
pLS   30
pLSs  100
pSS  100
pSSs  200
dm   20
dc   20
end parameters

begin species
Lig(l,l,l) Lig_tot
Lyn(U,SH2) Lyn_tot
Syk(tSH2,l~Y,a~Y) Syk_tot
Rec(a,b~Y,g~Y) Rec_tot
end species

begin reaction_rules
# Free ligand - receptor binding
1 Rec(a) + Lig(l,l,l) <-> Rec(a!1).Lig(l!1,l,l) kp1, km1

# Receptor-aggregation (ligand is already bound at one site)
2 Rec(a) + Lig(l,l,l!+) <-> Rec(a!1).Lig(l!1,l,l!+) kp2, km2

# Receptor-aggregation (ligand is already bound at two sites)
3 Rec(a) + Lig(l,l!+,l!+) <-> Rec(a!1).Lig(l!1,l!+,l!+) kp2, km2

# Constitutive Lyn-receptor binding
4 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL

# Transphosphorylation of beta by constitutive Lyn
5 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~Y) -> \\
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,b~pY) pLb
```

```
# Transphosphorylation of gamma by constitutive Lyn
6 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~Y) -> \\
Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY) pLg

# Lyn-receptor binding through SH2 domain
7 Rec(b~pY) + Lyn(U,SH2) <-> Rec(b~pY!1).Lyn(U,SH2!1) kpLs, kmLs

# Transphosphorylation of beta by SH2-bound Lyn
8 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~Y) -> \\
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,b~pY) pLbs

# Transphosphorylation of gamma by SH2-bound Lyn
9 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~Y) -> \\
Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY) pLgs

# Syk-receptor binding through tSH2 domain
10 Rec(g~pY) + Syk(tSH2) <-> Rec(g~pY!1).Syk(tSH2!1) kpS, kmS

# Transphosphorylation of Syk by constitutive Lyn
11 Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \\
   Lig(l!1,l!2).Lyn(U!3,SH2).Rec(a!2,b~Y!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLS

# Transphosphorylation of Syk by SH2-bound Lyn
12 Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~Y) -> \\
   Lig(l!1,l!2).Lyn(U,SH2!3).Rec(a!2,b~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,l~pY) pLSs

# Transphosphorylation of Syk by Syk not phosphorylated on aloop
13 Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \\
   Lig(l!1,l!2).Syk(tSH2!3,a~Y).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSS

# Transphosphorylation of Syk by Syk phosphorylated on aloop
14 Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~Y) -> \\
   Lig(l!1,l!2).Syk(tSH2!3,a~pY).Rec(a!2,g~pY!3).Rec(a!1,g~pY!4).Syk(tSH2!4,a~pY) pSSs

# Dephosphorylation of Rec beta
15 Rec(b~pY)-> Rec(b~Y) dm

# Dephosphorylation of Rec gamma
16 Rec(g~pY)-> Rec(g~Y) dm

# Dephosphorylation of Syk at membrane
17 Syk(tSH2!1,l~pY)-> Syk(tSH2!1,l~Y) dm
18 Syk(tSH2!1,a~pY)-> Syk(tSH2!1,a~Y) dm

# Dephosphorylation of Syk in cytosol
19 Syk(tSH2,l~pY)-> Syk(tSH2,l~Y) dc
20 Syk(tSH2,a~pY)-> Syk(tSH2,a~Y) dc

end reaction_rules
```

```
begin observables
Species       Ligfree     Lig(l,l,l)
Species       RecMon      Rec(a) Rec(a!1).Lig(l!1,l,l)
Species       RecDim      Rec(a!1).Lig(l!1,l!2,l).Rec(a!2)
Species       RecTri      Rec(a!1).Lig(l!1,l!2,l!3).Rec(a!2).Rec(a!3)
Molecules     RecPbeta    Rec(b~pY!?)
Molecules     RecPgamma   Rec(g~pY!?)
Molecules     RecSykPS    Rec(g~pY!1).Syk(tSH2!1,a~pY)
end observables

generate_network({overwrite=>1});
EOF

mkdir $prefix ;
chdir $prefix;

$fname = sprintf "${prefix}.bngl";
open(BNGL,">$fname") || die "Couldn't write to $fname" ;
print BNGL $script ;
$irun = 0 ;
foreach $i(@lig_conc){
  $irun += 1;
  $srun = sprintf "%6.5e",$i;
  if ($irun > 1){
    print BNGL "resetConcentrations()\n";
  }
  printf BNGL "setParameter(Lig_tot,$i);\n";
  printf BNGL "simulate_ode({suffix=>\"$srun\",t_end=>1000000,n_steps=>50000, \\
  atol=>1e-11,rtol=>1e-11,steady_state=>1});\n";
}
close(BNGL);

# Run BioNetGen on file
print "Running BioNetGen on $fname\n";
system("${BNGPATH}/Perl2/BNG2.pl $fname > $prefix.log");
```

E. How to choose L_T values for the bivalent ligand so that $\log_{10}(2K_1L)$ is a desired value?

In order to achieve this goal we need an expression relating L_T to $2K_1L$. The single site equilibrium constants for binding and cross-linking are $K_1=k_{+1}/k_{-1}$ and $K_2=k_{+2}/k_{-2}$ respectively. The concentrations at equilibrium of free receptor and receptor dimmers are denoted by R and D respectively. A 1:1 ligand receptor complex, in which one site on the bivalent ligand is bound to a receptor but the other site is free, is denoted by B. If there are RT total receptors on the cell surface and there are no processes to remove or add new receptors, the total number of receptors is conserved and

$$R_T = R + B + 2D = R + 2K_1LR + 2(2K_1L)(K_2/2)R^2$$
$$= (1+2K_1L)R + 2K_1K_2LR^2 \qquad (1)$$

Defining $r = R/R_T$ as the fraction of free receptors, the above equation reduces to

$$1 = (1+2K_1L)r + 2K_1K_2R_TLr^2 \quad (2)$$

It is straightforward to show that w, the fraction of receptors not in dimers, is given by

$$w = (1+2K_1L)r = (-1 + \sqrt{(1+4\delta)})/2\delta, \quad (3)$$

$$\text{where } \delta = (2K_1LK_2R_T)/(1+2K_1L)^2 \quad (4)$$

It follows that

$$R = wR_T/(1+2K_1L);\ B = wR_T(2K_1L)/(1+2K_1L);\ D = (1-w)R_T/2 \quad (5)$$

$$\text{and } L_T = L*V_{cell}*f + (B+D)*A_{cell}$$
$$= (2K_1L)*V_{cell}*f/2K_1 + (B+D)*A_{cell} \quad (6)$$

where f is a conversion factor from the given units of ligand concentration (L) to molecules per cell. For example, if L is in units of M (molar), then

$$f = N_{Av} *10^{-3} \quad (7)$$

where N_{Av} = Avogadro Number.

A Mathematica notebook for the calculation of L_T values for bivalent ligands, given the desired $2K_1L$ values is provided as a separate file named "convl2ltot_km10_kp1_1p6M_kp2RT_1.nb"

References

1. Faeder, J.R., et al., *Investigation of early events in Fc epsilon RI-mediated signaling using a detailed mathematical model.* J Immunol, 2003. **170**(7): p. 3769-81.