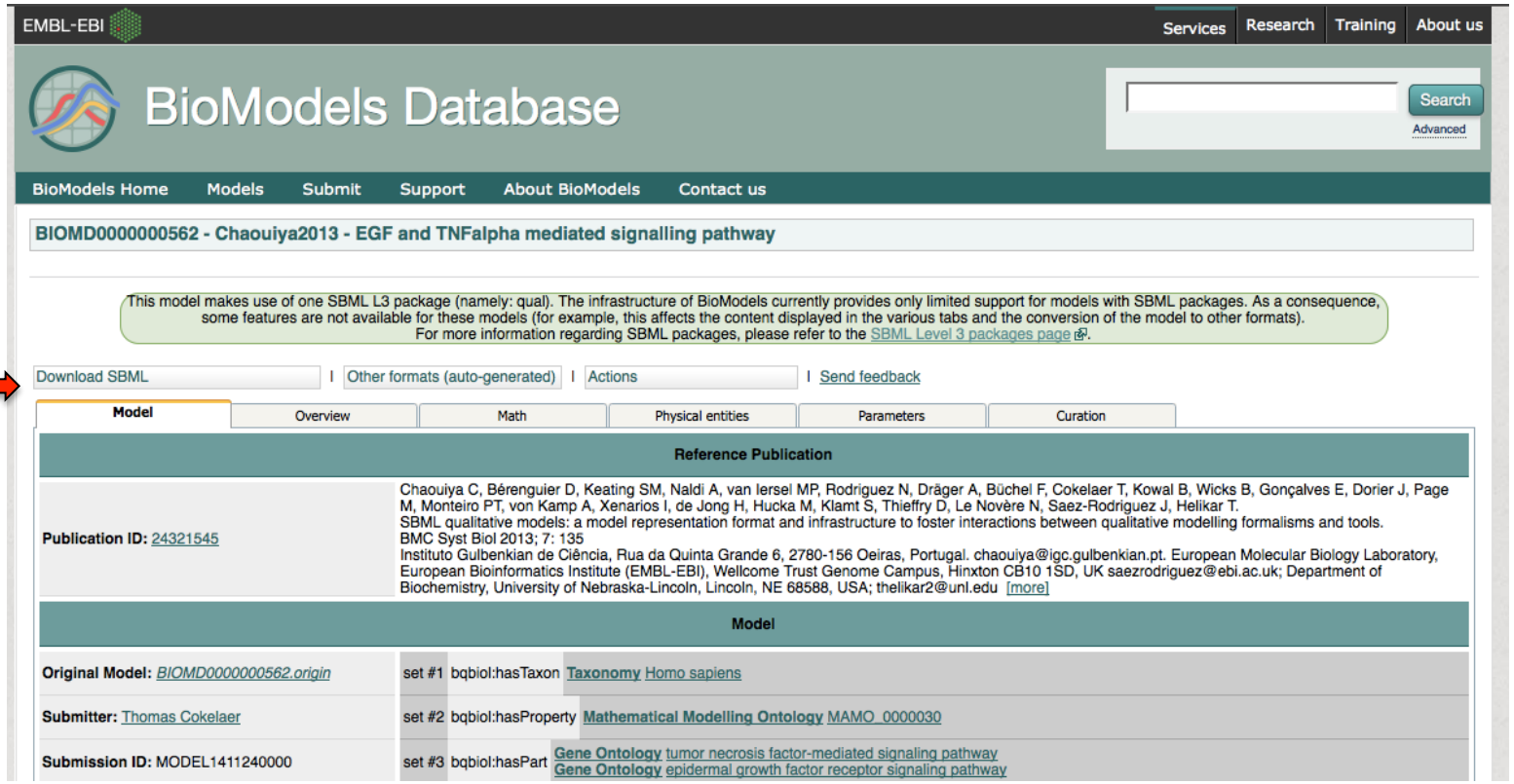
Example of MaBoSS use

Download a model from BioModels

- https://www.ebi.ac.uk/biomodels-main/BIOMD0000000562



- Download SBML (L3 V1)

institutCurie

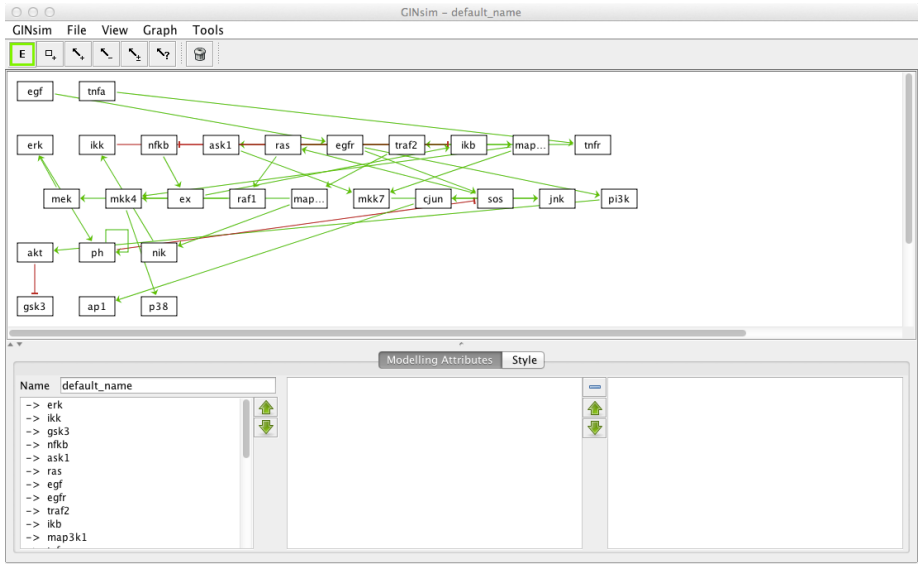
Inserm
Institut national
de la santé et de la recherche médicale

MINES PARIS

Open the model in GINsim

- Open latest version of GINsim

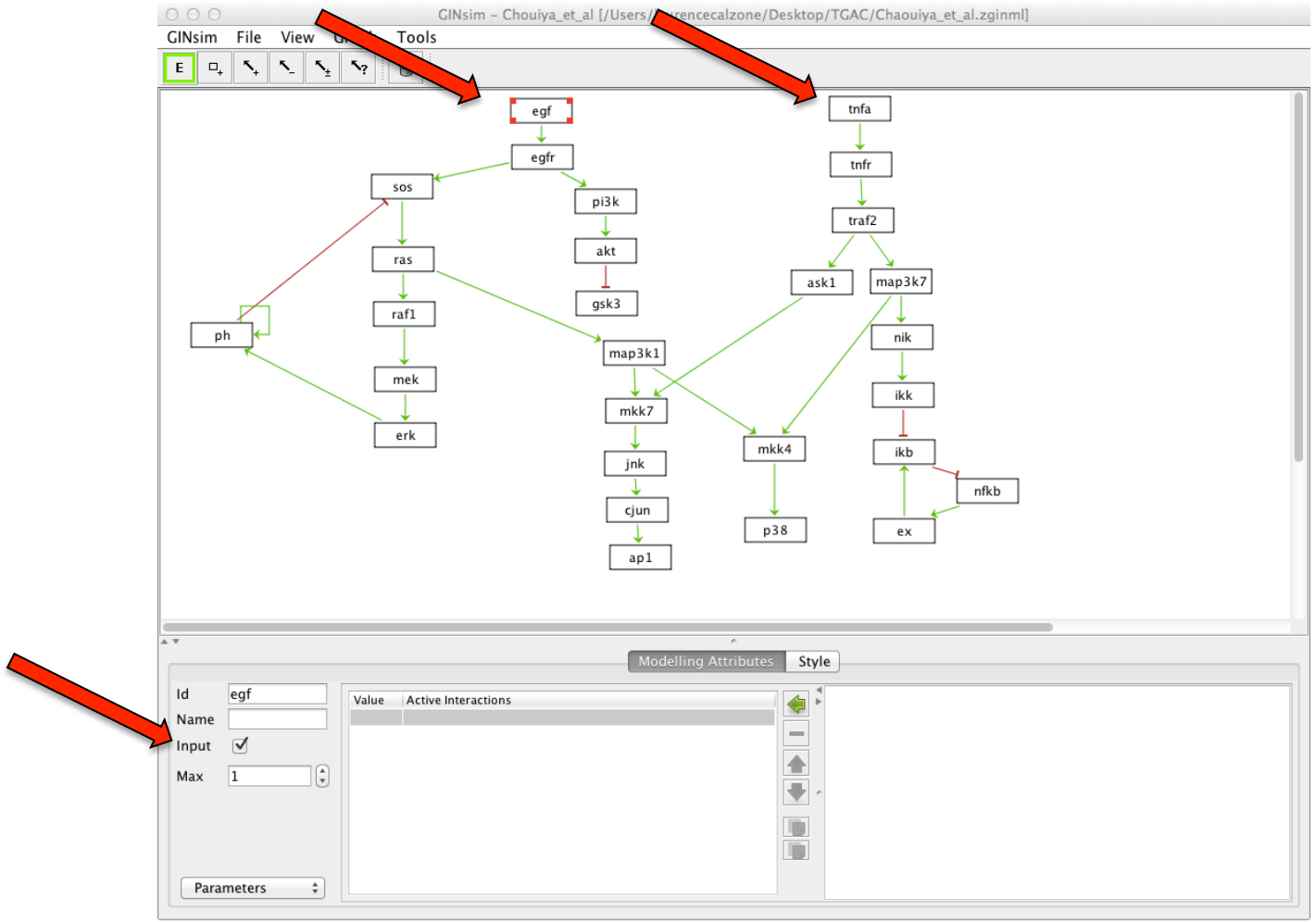
- Import => SBML-Qual
- Choose BIOMD0000000562.xls



- Adapt the layout
  (or open Chaouiya_et_al.zginml by selecting File => Open)

Simulate a model

Select egf and tnfa separately and click on input*

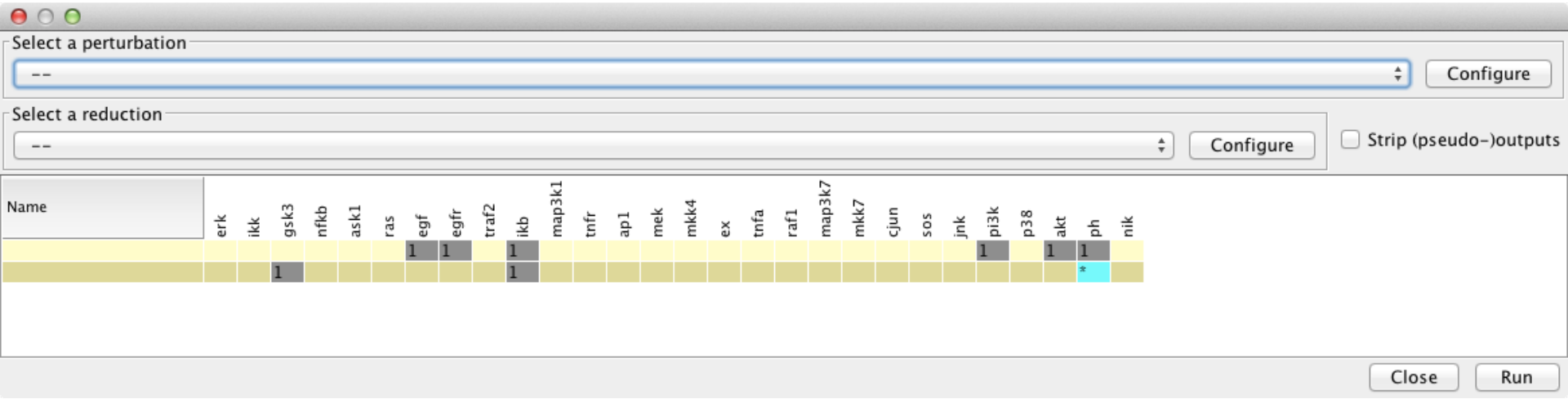


* done for you in Chaouiya_et_al.zginml

Compute stable state solutions of the wild type

Tools => Compute stable states

Click on Run



Export to MaBoSS

File => Export => MaBoSS

- Choose name of the file “chaouiya_maboss”
- Two files are created:

chaouiya_maboss.bnd

chaouiya_maboss.cfg

chaouiya_maboss.bnd

```
Node erk {
  logic = (mek);
  rate_up = @logic ? $u_erk : 0;
  rate_down = @logic ? 0 : $d_erk;
}

Node ikk {
  logic = (nik);
  rate_up = @logic ? $u_ikk : 0;
  rate_down = @logic ? 0 : $d_ikk;
}

Node gsk3 {
  logic = (!akt);
  rate_up = @logic ? $u_gsk3 : 0;
  rate_down = @logic ? 0 : $d_gsk3;
}

Node nfkb {
  logic = (!ikb);
  rate_up = @logic ? $u_nfkb : 0;
  rate_down = @logic ? 0 : $d_nfkb;
}

Node ask1 {
  logic = (traf2);
  rate_up = @logic ? $u_ask1 : 0;
  rate_down = @logic ? 0 : $d_ask1;
```

chaouiya_maboss.cfg

```
tnfr.is_internal=1;
ap1.is_internal=1;
mek.is_internal=1;
mkk4.is_internal=1;
ex.is_internal=1;
tnfa.is_internal=1;
raf1.is_internal=1;
map3k7.is_internal=1;
mkk7.is_internal=1;
cjun.is_internal=1;
sos.is_internal=1;
jnk.is_internal=1;
pi3k.is_internal=1;
p38.is_internal=1;
akt.is_internal=0;
ph.is_internal=1;
nik.is_internal=1;


discrete_time = 0;
use_physrandgen = FALSE;
seed_pseudorandom = 100;
sample_count = 50000;

max_time = 5;
time_tick = 0.01;

thread_count = 4;

statdist_traj_count = 100;
statdist_cluster_threshold = 0.9;
```

Definition of the logical rules

Definition of the simulation parameters

Install MaBoSS

- Install MaBoSS

$ **cd MaBoSS-2.0-env/src**

$ **make install**

- gcc: version 4.0.1 or higher
  - bison: version 2.3 or higher
  - flex: version 2.5.35 or higher

- Set the environement variables for using scripts by command line:

$ **cd MaBoSS-2.0-env/**

$ **source ./MaBoSS.env**

Create mutant bnd file

- In bnd file, add parameters to simulate mutations:

  For KO:

  rate_up = $node_ko ? 0.0 : (@logic ? 1 ; 0);

  rate_down = $node_ko ? max_rate : (@logic ? 0 ; 1);

  For overexpression:

  rate_up = $node_up ? max_rate : ? (@logic ? 1 ; 0);

  rate_down = $node_up ? 0.0 : (@logic ? 0 ; 1);

- Or you can generate the file automatically.

In a terminal, type:

$ MBSS_MutBnd.pl chaouiya_maboss.bnd "akt erk ras pi3k"

- Verify that the nodes are recognized

Catch node erk

Catch node ras

Catch node pi3k

Catch node akt

Create mutant cfg file

- A new bnd files is created:

chaouiya_maboss_mut.bnd

- Create the corresponding cfg file

$ MaBoSS -t chaouiya_maboss_mut.bnd > chaouiya_maboss_mut.cfg

Or use directly the command:

$ MBSS_MutBndCfg.pl chaouiya_maboss.bnd "akt erk ras pi3k"

```
//
// MaBoSS 1.3.8 configuration template generated at Fri Jan 15 15:04:12
2016
//

// global configuration variables
time_tick = 0.5;
max_time = 1000;
sample_count = 10000;
discrete_time = 0;
use_physrandgen = 0;
seed_pseudorandom = 0;
display_traj = 0;
statdist_traj_count = 0;
statdist_cluster_threshold = 1;
thread_count = 1;
statdist_similarity_cache_max_size = 20000;

// variables to be set in the configuration file or by using the --
config-vars option
$akt_ko = 0;
$akt_up = 0;
$d_akt = 0;
$d_ap1 = 0;
$d_ask1 = 0;
$d_cjun = 0;
$d_egf = 0;
$d_egfr = 0;
$d_erk = 0;
$d_ex = 0;
$d_gsk3 = 0;
$d_ikb = 0;
$d_ikk = 0;
$d_jnk = 0;
$d_map3k1 = 0;
$d_map3k7 = 0;
$d_mek = 0;
$d_mkk4 = 0;
$d_mkk7 = 0;
$d_nfkb = 0;
$d_nik = 0;
$d_p38 = 0;
$d_ph = 0;
$d_pi3k = 0;
$d_raf1 = 0;
$d_ras = 0;
$d_sos = 0;
$d_tnfa = 0;
$d_tnfr = 0;
$d_traf2 = 0;
$erk_ko = 0;
$erk_up = 0;
$pi3k_ko = 0;
$pi3k_up = 0;
$ras_ko = 0;
$ras_up = 0;
```

time_tick = 0.5 => 0.1
max_time = 1000 => 60
sample_count = 10000 => 50000

All parameters for mutant definition need to be set to 0 (default)

```
// set is_internal attribute value to 1 if node is an internal
node
mek.is_internal = 0;
erk.is_internal = 0;
nik.is_internal = 0;
ikk.is_internal = 0;
akt.is_internal = 0;
gsk3.is_internal = 0;
ikb.is_internal = 0;
nfkb.is_internal = 0;
traf2.is_internal = 0;
ask1.is_internal = 0;
sos.is_internal = 0;
ras.is_internal = 0;
egf.is_internal = 0;
egfr.is_internal = 0;
tnfr.is_internal = 0;
ex.is_internal = 0;
map3k1.is_internal = 0;
tnfa.is_internal = 0;
cjun.is_internal = 0;
ap1.is_internal = 0;
raf1.is_internal = 0;
map3k7.is_internal = 0;
mkk4.is_internal = 0;
mkk7.is_internal = 0;
jnk.is_internal = 0;
ph.is_internal = 0;
pi3k.is_internal = 0;
p38.is_internal = 0;
```

all variables need to be set internal (=1) except for the ones you wish to see in the output (readout)

erk.is_internal=0;
akt.is_internal=0;
ras.is_internal=0;

```
// if NODE initial state is:
// - equals to 1: NODE.istate = 1;
// - equals to 0: NODE.istate = 0;
// - random: NODE.istate = -1; OR [NODE].istate = 0.5 [0], 0.5
[1]; OR skip NODE.istate declaration
// - weighted random: [NODE].istate = P0 [0], P1 [1]; where P0
and P1 are arithmetic expressions
[mek].istate = 0.5 [0], 0.5 [1];
[erk].istate = 0.5 [0], 0.5 [1];
[nik].istate = 0.5 [0], 0.5 [1];
[ikk].istate = 0.5 [0], 0.5 [1];
[akt].istate = 0.5 [0], 0.5 [1];
[gsk3].istate = 0.5 [0], 0.5 [1];
[ikb].istate = 0.5 [0], 0.5 [1];
[nfkb].istate = 0.5 [0], 0.5 [1];
[traf2].istate = 0.5 [0], 0.5 [1];
[ask1].istate = 0.5 [0], 0.5 [1];
[sos].istate = 0.5 [0], 0.5 [1];
[ras].istate = 0.5 [0], 0.5 [1];
[egf].istate = 0.5 [0], 0.5 [1];
[egfr].istate = 0.5 [0], 0.5 [1];
[tnfr].istate = 0.5 [0], 0.5 [1];
[ex].istate = 0.5 [0], 0.5 [1];
[map3k1].istate = 0.5 [0], 0.5 [1];
[tnfa].istate = 0.5 [0], 0.5 [1];
[cjun].istate = 0.5 [0], 0.5 [1];
[ap1].istate = 0.5 [0], 0.5 [1];
[raf1].istate = 0.5 [0], 0.5 [1];
[map3k7].istate = 0.5 [0], 0.5 [1];
[mkk4].istate = 0.5 [0], 0.5 [1];
[mkk7].istate = 0.5 [0], 0.5 [1];
[jnk].istate = 0.5 [0], 0.5 [1];
[ph].istate = 0.5 [0], 0.5 [1];
[pi3k].istate = 0.5 [0], 0.5 [1];
[p38].istate = 0.5 [0], 0.5 [1];
```

initial conditions.

By default = all random

Modify such that:
[tnfa].istate = 1 [0], 0 [1]
[erk].istate = 1 [0], 0 [1]
[akt].istate = 1 [0], 0 [1]
[ras].istate = 1 [0], 0 [1]

Done for you in folder

Simulate the wild type with MaBoSS

- Type in a terminal the command:

$ MBSS_FormatTable.pl chaouiya_maboss_mut.bnd chaouiya_maboss_mut.cfg

- Visualize the results
  - For each simulation, a new folder is created
  - In this folder, open chaouiya_maboss_mut_fp.csv
  - 3 stable states are found with the probabilities associated to each of them

FP	Proba	State	mek	erk	nik	ikk	akt	gsk3	ikb	nfkb	traf2	ask1	sos	ras	egf	egfr	tnfr	ex	map3k1	tnfa	cjun	ap1	raf1	map3k7	mkk4	mkk7	jnk	ph	pi3k	p38
#1	0.5003	akt -- ikb -- egf -- egfr -- ph -- pi3k	0	0	0	0	1	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
#2	0.37334	gsk3 -- ikb -- ph	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
#3	0.12636	gsk3 -- ikb	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Stable states for the wild-type

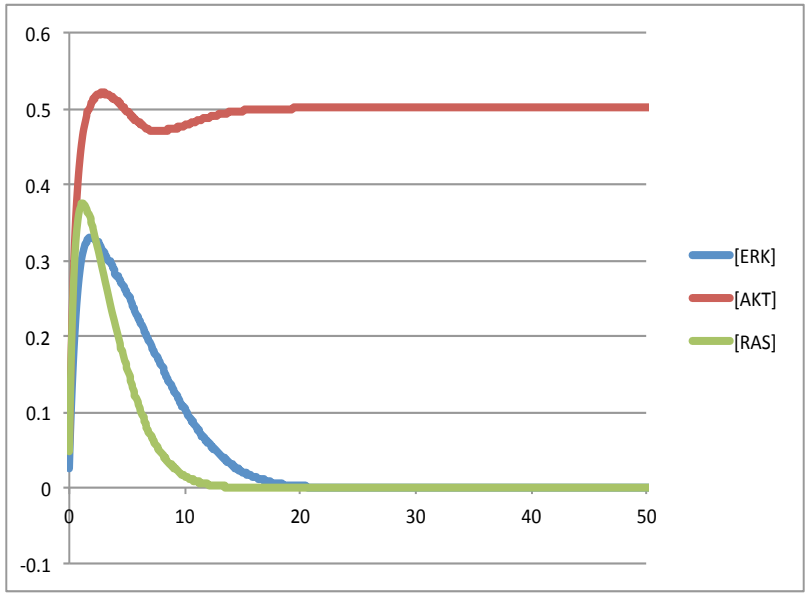
egf	tnfa	akt	ap1	ask1	cjun	egfr	erk	ex	gsk3	ikb	ikk	jnk	map3k1	map3k7	mek	mkk4	mkk7	nfkb	nik	p38	ph	pi3k	raf1	ras	sos	tnfr	traf2
0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0

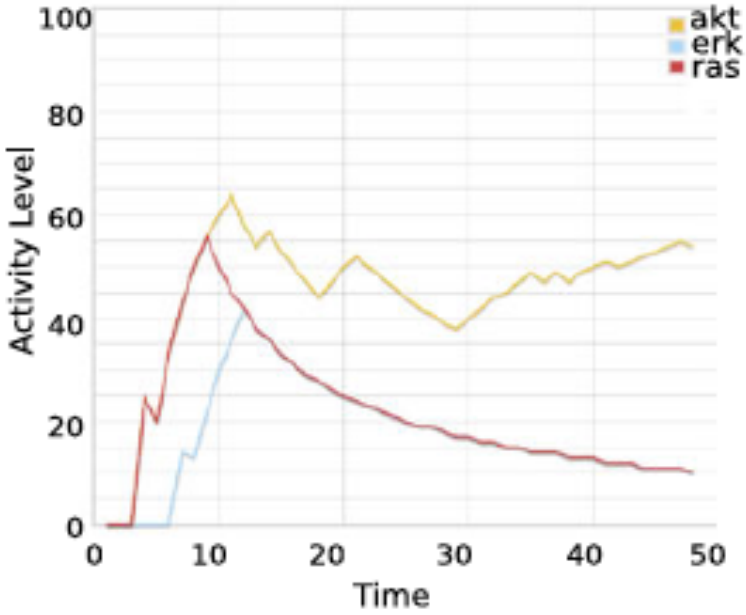
institutCurie

Inserm
Institut national de la santé et de la recherche médicale

MINES PARIS

- In this folder, open chaouiya_maboss_mut_probtraj_table.csv





→ plot the sum of the three outputs

→ verify with the initial article

NB: to compare with publication, compute sum of each component:
[ERK]=sum(erk)=prob[erk]+prob[erk-ras]+prob[erk-akt]+prob[erk-ras-akt]

Simulate a mutant with MaBoSS

In cfg file, set parameters to simulate ras gain of function mutations:

$ras_ko=0; (not modified)

$ras_up=1;

[ras].istate = 0 [0], 1 [1];

Save the cfg file as: ras_oe.cfg

Simulate mutant in MaBoSS:

$ MBSS_FormatTable.pl chaouiya_maboss_mut.bnd ras_oe.cfg

FP	Proba	mek	erk	nik	ikk	akt	gsk3	ikb	nfkb	traf2	ask1	sos	ras	egf	egfr	tnfr	ex	map3k1	tnfa	cjun	ap1	raf1	map3k7	mkk4	mkk7	jnk	ph	pi3k	p38
#1	0.5003	1	1	0	0	1	0	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	0	1	1	1	1	0
#2	0.4997	1	1	0	0	0	1	1	0	0	0	0	1	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	0

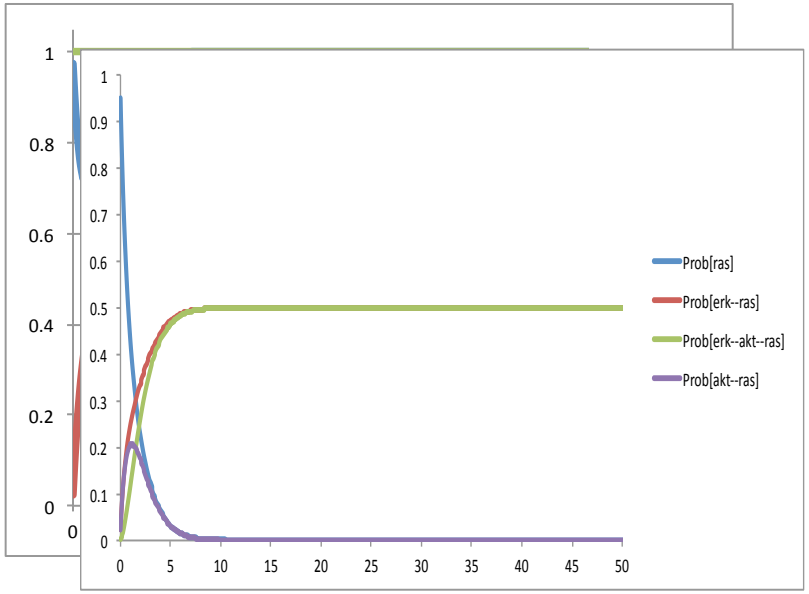
Three ways to visualize the results

1. Stable states:

FP	Proba	mek	erk	nik	ikk	akt	gsk3	ikb	nfkb	traf2	ask1	sos	ras	egf	egfr	tnfr	ex	map3k1	tnfa	cjun	ap1	raf1	map3k7	mkk4	mkk7	jnk	ph	pi3k	p38
#1	0.5003	1	1	0	0	1	0	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	0	1	1	1	1	0
#2	0.4997	1	1	0	0	0	1	1	0	0	0	0	1	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	0

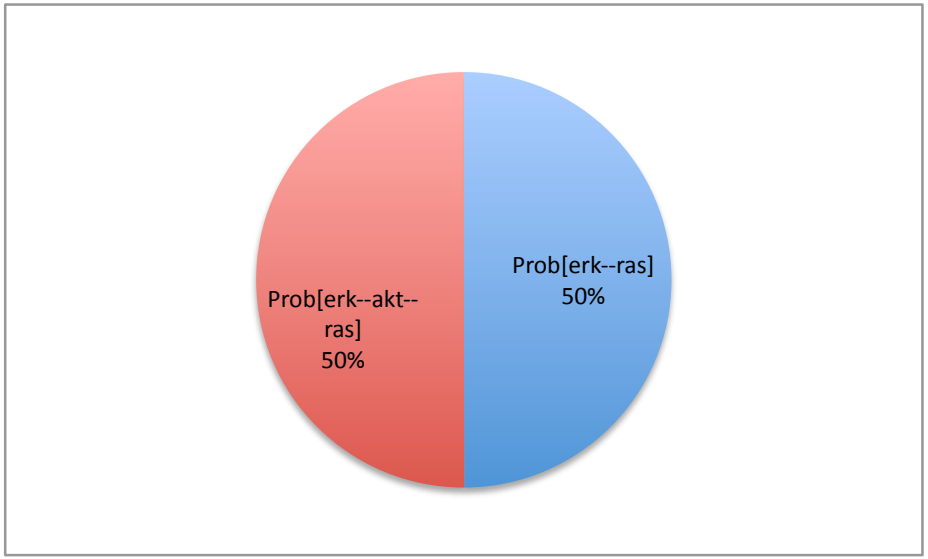
useful to verify with GINsim, for instance

2. Evolution over time:



interesting if transient behaviours

3. distribution of asymptotic solutions



interesting if presence of limit cycles

institutCurie

Inserm
Institut national
de la santé et de la recherche médicale

MINES PARIS