

Examples of published models

In this notebook, we show examples of previously published models encoded using [Combine archives](#). The point of Combine archive handling in Tellurium is **not** to comprehensively reproduce all simulations in a manuscript. Instead, the point is to provide a starting point, e.g. a single simulation, that allows a user to import a model and validate that it is working. The user can then build more complex analyses on top of this starting point.

Usage

To run any of these examples, select the cell and press Ctrl+Enter. To export the cell as a Combine archive, select the cell and click the save icon (diskette) in the upper-right.

Example 1: Metabolic Oscillations in Yeast

Reference:

Wolf, J., Sohn, H. Y., Heinrich, R., & Kuriyama, H. (2001). [Mathematical analysis of a mechanism for autonomous metabolic oscillations in continuous culture of Saccharomyces cerevisiae](#). *FEBS letters*, 499(3), 230-234.

Example 2: EGF/Insulin Signaling Crosstalk

Reference:

Borisov, N., Aksamitiene, E., Kiyatkin, A., Legewie, S., Berkhout, J., Maiwald, T., ... & Kholodenko, B. N. (2009). [Systems-level interactions between insulin-EGF networks amplify mitogenic signaling](#). *Molecular systems biology*, 5(1), 256.

Example 3: p53 Signaling Models

Reference:

Proctor, C. J., & Gray, D. A. (2008). [Explaining oscillations and variability in the p53-Mdm2 system](#). *BMC systems biology*, 2(1), 75.

Example 1: Metabolic Oscillations in Yeast (Wolf, 2001)

This example shows respiratory oscillations in yeast. It was the first mechanistic model of this process and is part of the curated branch of the Biomed repository. The first plot corresponds to Figure 2B from the original paper (albeit with a slight phase shift as noted in the source model below). The second plot shows a phase portrait, which, although not part of the original paper, fits nicely with the limit cycle oscillations present in this model.

Source model: [BIOMD000000090](#)

Reference:

Wolf, J., Sohn, H. Y., Heinrich, R., & Kuriyama, H. (2001). [Mathematical analysis of a mechanism for autonomous metabolic oscillations in continuous culture of Saccharomyces cerevisiae](#). *FEBS letters*, 499(3), 230-234.

[Back to top](#)

```
// -- Begin Antimony block converted from wolf2001.xml
// Created by libAntimony v2.9.1
model *wolf2001()

// Compartments and Species:
compartment c0, c1, c2;
species $sul_ex in c0, $eth_ex in c0, $oxy_ex in c0, oxy in c2, $H2O in c2;
species A3c in c1, aps in c1, $PPi in c1, pap in c1, sul in c1, eth in c1;
species $A2c in c1, hyd in c1, cys in c1, N2 in c1, $N1 in c1, aco in c1;
species oah in c1, S1 in c2, $S2 in c2, $C1 in c2, $C2 in c2, $A2m in c2;
species A3m in c2, $Ho in c1, $Hm in c2;

// Assignment Rules:
A2c := Ac - A3c;
N1 := N - N2;
S2 := S - S1;
A2m := Am - A3m;

// Reactions:
v1: $sul_ex => sul; c0*k_v0/(1 + (cys/Kc)^n);
v13: $eth_ex => eth; c0*k_v13;
v2: sul + A3c => aps + $PPi; c1*k2*sul*A3c;
v10: $oxy_ex => oxy; c0*k_v10;
v14: oxy => $oxy_ex; c2*k14*oxy;
v3: aps + A3c => pap + $A2c; c1*k3*aps*A3c;
v4: pap + 3N2 => hyd + 3$N1; c1*k4*pap*N2;
v5: hyd + oah => cys; c1*k5*hyd*oah;
v6: cys => ; c1*k6*cys;
v7: eth + 2$N1 => aco + 2N2; c1*k7*eth*N1;
v15: aco => oah; c1*k15*aco;
v17: hyd => ; c1*k17*hyd;
v18: oah => ; c1*k18*oah;
v8: $S2 + aco => S1; c2*k8*aco*S2;
v9: S1 + 4$N1 => $S2 + 4N2; c2*k9*S1*N1;
v11a: $C1 + $Hm + N2 => $C2 + $Ho + $N1; c2*k11*N2*oxy/((a*N2 + oxy)*(1 + (hyd/Kh)^m));
v11a2: $C2 + oxy => $C1 + $H2O; c2*k11*N2*oxy/((a*N2 + oxy)*(1 + (hyd/Kh)^m));
v16: $A2c + A3m => $A2m + A3c; c2*k16*A3m*A2c;
v11b: $Ho + $A2m => $Hm + A3m; (c2*3*k11*N2*oxy/((a*N2 + oxy)*(1 + (hyd/Kh)^m)))*A2m/(Ka + A2m);
vLEAK: $Ho => $Hm; 0;
v12: A3c => $A2c; c1*k12*A3c;

// Species initializations:
sul_ex = 0;
eth_ex = 0;
oxy_ex = 0;
oxy = 7;
oxy has substance_per_volume;
H2O = 0;
A3c = 1.5;
A3c has substance_per_volume;
aps = 0.5;
aps has substance_per_volume;
PPi = 0;
pap = 0.4;
pap has substance_per_volume;
sul = 0.4;
sul has substance_per_volume;
eth = 4;
eth has substance_per_volume;
A2c has substance_per_volume;
hyd = 0.5;
hyd has substance_per_volume;
cys = 0.3;
cys has substance_per_volume;
N2 = 2;
N2 has substance_per_volume;
N1 has substance_per_volume;
aco = 0.3;
aco has substance_per_volume;
oah = 1.5;
oah has substance_per_volume;
S1 = 1.5;
S1 has substance_per_volume;
S2 has substance_per_volume;
C1 = 0;
C2 = 0;
A2m has substance_per_volume;
A3m = 1.5;
A3m has substance_per_volume;
Ho = 0;
Hm = 0;

// Compartment initializations:
c0 = 1;
c1 = 1;
c2 = 1;

// Variable initializations:
Ac = 2;
N = 2;
S = 2;
Am = 2;
k_v0 = 1.6;
Kc = 0.1;
n = 4;
k_v13 = 4;
k2 = 0.2;
k_v10 = 80;
k14 = 10;
k3 = 0.2;
k4 = 0.2;
k5 = 0.1;
k6 = 0.12;
k7 = 10;
k15 = 5;
k17 = 0.02;
k18 = 1;
k8 = 10;
k9 = 10;
k11 = 10;
a = 0.1;
Kh = 0.5;
m = 4;
k16 = 10;
Ka = 1;
k12 = 5;

// Other declarations:
const c0, c1, c2, Ac, N, S, Am, k_v0, Kc, n, k_v13, k2, k_v10, k14, k3;
const k4, k5, k6, k7, k15, k17, k18, k8, k9, k11, a, Kh, m, k16, Ka, k12;

// Unit definitions:
unit substance_per_volume = mole / litre;
unit substance = mole;

// Display Names:
c0 is "external";
c1 is "cytosol";
c2 is "mitochondria";
sul_ex is "S04_ex";
eth_ex is "EtOH_ex";
oxy_ex is "O2_ex";
oxy is "O2";
A3c is "ATP";
aps is "APS";
pap is "PAPS";
sul is "S04";
eth is "EtOH";
A2c is "ADP";
hyd is "H2S";
cys is "CYS";
N2 is "NADH";
N1 is "NAD";
aco is "AcCoA";
oah is "OAH";
A2m is "ADP_mit";
A3m is "ATP_mit";
v11a is "vET1";
v11a2 is "vET2";
v11b is "vSYNT";
end
// -- End Antimony block

// -- Begin PhraSEDML block converted from main.xml
// Created by libPhraSEDML v1.0.5
// Models
model1 = model "wolf2001"

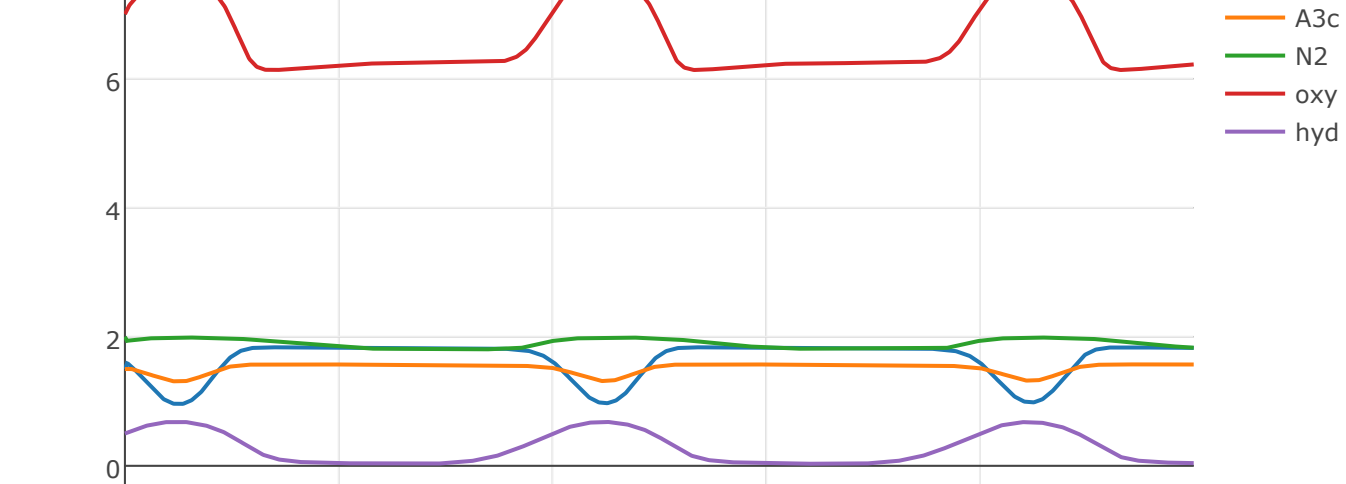
// Simulations
sim1 = simulate uniform(0, 100, 1000)

// Tasks
task1 = run sim1 on model1

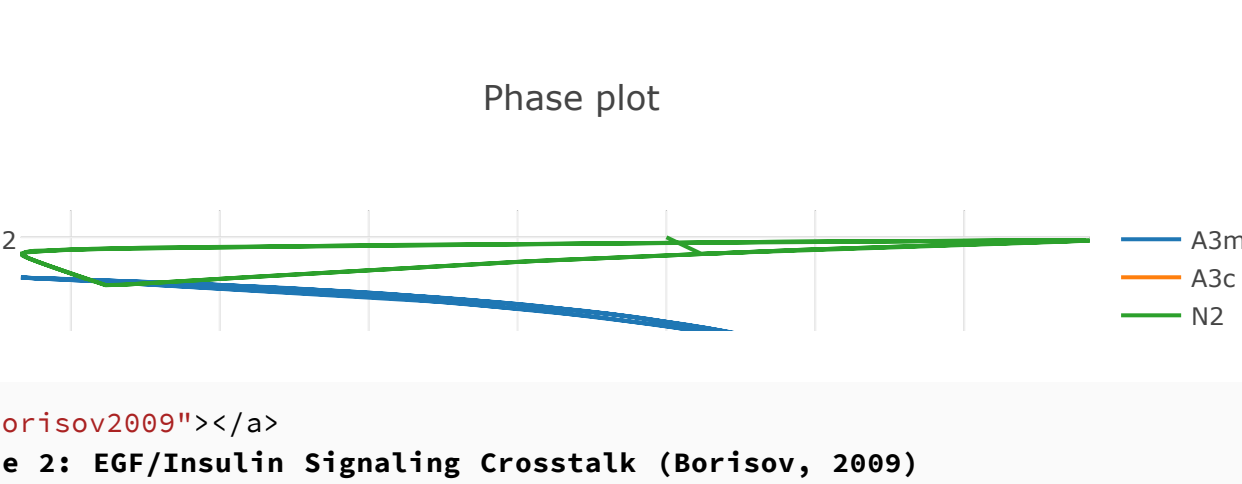
// Outputs
plot "Limit cycle oscillations (Wolf, 2001)" time vs A3m, A3c, N2, oxy, hyd

plot "Phase plot" oxy vs A3m, A3c, N2, hyd
// -- End PhraSEDML block
```

Limit cycle oscillations (Wolf, 2001)



Phase plot



```
<a id="borisov2009"></a>
# Example 2: EGF/Insulin Signaling Crosstalk (Borisov, 2009)

In this example, both Erk and Akt can be either singly- or doubly-phosphorylated. We show how to plot the total of both forms in order to illustrate how to plot derived quantities. In the original manuscript, only the doubly-phosphorylated form was plotted.

## Source model: \[BIOMD000000223\](https://www.ebi.ac.uk/biomodels-main/BIOMD0000000223)

## Reference:

Borisov, N., Aksamitiene, E., Kiyatkin, A., Legewie, S., Berkhout, J., Maiwald, T., ... & Kholodenko, B. N. (2009). \[Systems-level interactions between insulin-EGF networks amplify mitogenic signaling\](http://msb.embopress.org/content/5/1/256) Molecular systems biology, 5(1), 256.

### \[Back to top\](#topcell)
```

Example 2: EGF/Insulin Signaling Crosstalk (Borisov, 2009)

In this example, both Erk and Akt can be either singly- or doubly-phosphorylated. We show how to plot the total of both forms in order to illustrate how to plot derived quantities. In the original manuscript, only the doubly-phosphorylated form was plotted.

Source model: [BIOMD000000223](#)

Reference:

Borisov, N., Aksamitiene, E., Kiyatkin, A., Legewie, S., Berkhout, J., Maiwald, T., ... & Kholodenko, B. N. (2009). [Systems-level interactions between insulin-EGF networks amplify mitogenic signaling](#). Molecular systems biology, 5(1), 256.

Back to top

```
// -- Begin Antimony block converted from borisov2009.xml
// Created by libAntimony v2.9.1
model *borisov2009()

    // Compartments and Species:
    compartment cell, extra;
    species EGF in extra, I in extra, RE in cell, Rd in cell, Rp in cell, Rp_GS in cell;
    species Rp_GS in cell, Shc in cell, Rp_Shc in cell, Rp_pShc in cell, pShc in cell;
    species Rp_pShc_GS in cell, PI3K in cell, Rp_PI3K in cell, RasGAP in cell;
    species Rp_RasGAP in cell, IRL in cell, IRp in cell, IRp_PI3K in cell, IRp_RasGAP in
cell;
    species IRS in cell, IRp_IRS in cell, IRp_IRSp in cell, IRSp in cell, iSrc in cell;
    species mIRS in cell, mIRSp in cell, mIRSp_GS in cell, mIRSp_PI3K in cell;
    species SHP2 in cell, mIRSp_SHP2 in cell, GAB in cell, mGAB in cell, mGABp in cell;
    species mGABp_GS in cell, mGABp_PI3K in cell, mGABp_SHP2 in cell, mGABp_pSHP2 in cell;
    species PIP3 in cell, dRas in cell, Raf in cell, aRaf in cell, Mek in cell;
    species Erk in cell, pErk in cell, PDK1 in cell, Akt in cell, pAkt in cell;
    species mTOR in cell, Null in cell, aaRaf in cell, PKA in cell, pShc_GS in cell;
    species ppMek in cell, mGABp_pSHP2_GS in cell, R in cell, ppErk in cell;
    species IR in cell, mPDK1 in cell, tRas in cell, tRas_PI3K in cell, ppAkt in cell;
    species mGABp_RasGAP in cell, amTOR in cell, iGS in cell, imGAB in cell;
    species imIRS in cell, aSrc in cell, Ri in cell, Iri in cell, iPX in cell;
    species aPX in cell, aPX_GS in cell, IRSp_PI3K in cell, IRSp_GS in cell;
    species IRSp_SHP2 in cell, GABp in cell, GABp_PI3K in cell, GABp_GS in cell;
    species GABp_RasGAP in cell, GABp_SHP2 in cell, GABp_pSHP2 in cell, GABp_pSHP2_GS in
cell;
    species imGABp in cell, bRasGAP in cell, $phosphorylated_Akt in cell;

    // Assignment Rules:
    phosphorylated_Akt := pAkt + ppAkt;
    EGF_tot := EGF + (RE + 2*(Rd + Rp + Ri + Rp_GS + Rp_Shc + Rp_pShc + Rp_pShc_GS +
Rp_PI3K + Rp_RasGAP))*(cell/extra);
    k11 := k9;
    k_1 := Kd1*k1;
    k_2 := Kd2*k2;
    k_4 := Kd4*k4;
    k_5 := Kd5*k5;
    k_7 := Kd7*k7;
    k_9 := Kd9*k9;
    k_10 := Kd10*k10;
    k_11 := k_9;
    k_12 := Kd12*k12;
    k_13 := Kd13*k13;
    k_24 := Kd24*k24;
    k_26 := Kd26*k26;
    k_27 := Kd27*k27;
    k_28 := Kd28*k28;
    k_30 := Kd30*k30;
    k_42 := Kd42*k42;
    k_45 := k45*Kd45;
    k_46 := Kd46*k46;
    k_47 := Kd47*k47;
    k_49 := Kd49*k49;
    k_52 := k52*Kd52;
    k_53 := Kd53*k53;
    k_54 := Kd54*k54;
    k_55 := Kd55*k55;
    k_59 := Kd59*k59;
    k_74 := k74*Kd74;

    // Reactions:
    reaction_1: R + EGF -> RE; (k1*R*EGF - k_1*RE)*cell;
    reaction_2: 2RE -> Rd; (k2*RE*RE - k_2*Rd)*cell;
    reaction_3: Rd => Rp; k3*Rd*cell;
    reaction_4: Rp + GS -> Rp_GS; (k4*Rp*GS - k_4*Rp_GS)*cell;
    reaction_5: Rp + Shc -> Rp_Shc; (k5*Rp*Shc - k_5*Rp_Shc)*cell;
    reaction_6: Rp_Shc => Rp_pShc; k6*Rp_Shc*cell;
    reaction_7: Rp_pShc -> Rp + pShc; (k_7*Rp_pShc - k7*Rp*pShc)*cell;
    reaction_8: pShc => Shc; (V8*pShc/(Km8 + pShc))*cell;
    reaction_9: GS + Rp_pShc -> Rp_pShc_GS; (k9*Rp_pShc*GS - k_9*Rp_pShc_GS)*cell;
    reaction_10: Rp_pShc_GS -> Rp + pShc_GS; (k_10*Rp_pShc_GS - k10*Rp*pShc_GS)*cell;
    reaction_11: pShc_GS -> GS + pShc; (k_11*pShc_GS - k11*pShc*GS)*cell;
    reaction_12: Rp + PI3K -> Rp_PI3K; (k12*Rp*PI3K - k_12*Rp_PI3K)*cell;
    reaction_13: Rp + RasGAP -> Rp_RasGAP; (k13*Rp*RasGAP - k_13*Rp_RasGAP)*cell;
    reaction_17: Rp => Null; k17*Rp*cell;
    reaction_18: Rp_GS => GS; k17*Rp_GS*cell;
    reaction_19: Rp_Shc => Shc; k17*Rp_Shc*cell;
    reaction_20: Rp_pShc => pShc; k17*Rp_pShc*cell;
    reaction_21: Rp_pShc_GS => pShc_GS; k17*Rp_pShc_GS*cell;
    reaction_22: Rp_PI3K => PI3K; k17*Rp_PI3K*cell;
    reaction_23: Rp_RasGAP => RasGAP; k17*Rp_RasGAP*cell;
    reaction_24: I + IR -> IRL; (k24*IR*I - k_24*IRL)*cell;
    reaction_25: IRL => IRp; k25*IRL*cell;
    reaction_26: PI3K + IRp -> IRp_PI3K; (k26*IRp*PI3K - k_26*IRp_PI3K)*cell;
    reaction_27: RasGAP + IRp -> IRp_RasGAP; (k27*IRp*RasGAP - k_27*IRp_RasGAP)*cell;
    reaction_28: IRp + IRS -> IRp_IRS; (k28*IRp*IRS - k_28*IRp_IRS)*cell;
    reaction_29: IRp_IRS => IRp_IRSp; k29*IRp_IRS*cell;
    reaction_30: IRp_IRSp -> IRp + IRSp; (k_30*IRp_IRSp - k30*IRp*IRSp)*cell;
    reaction_31: IRSp => IRS; (V31*IRSp/(Km31 + IRSp))*cell;
    reaction_34: IRp => Null; k17*IRp*cell;
    reaction_35: IRp_PI3K => PI3K; k17*IRp_PI3K*cell;
    reaction_36: IRp_RasGAP => RasGAP; k17*IRp_RasGAP*cell;
    reaction_37: IRp_IRS => IRS; k17*IRp_IRS*cell;
    reaction_38: IRp_IRSp => IRSp; k17*IRp_IRSp*cell;
    reaction_40: iSrc => aSrc; (kcat40*iSrc*(Rp + alpha40*IRp)/(Km40 + iSrc))*cell;
    reaction_41: aSrc => iSrc; (V41*aSrc/(Km41 + aSrc))*cell;
    reaction_42: IRS + PIP3 -> mIRS; (k42*IRS*PIP3 - k_42*mIRS)*cell;
    reaction_43: mIRS => mIRSp; (kcat43*mIRS*(IRp + alpha43*Rp)/(Km43 + mIRS))*cell;
    reaction_44: mIRSp => mIRS; (V31*mIRSp/(Km31 + mIRSp))*cell;
    reaction_45: GS + mIRSp -> mIRSp_GS; (k45*mIRSp*GS - k_45*mIRSp_GS)*cell;
    reaction_46: PI3K + mIRSp -> mIRSp_PI3K; (k46*mIRSp*PI3K - k_46*mIRSp_PI3K)*cell;
    reaction_47: mIRSp + SHP2 -> mIRSp_SHP2; (k47*mIRSp*SHP2 - k_47*mIRSp_SHP2)*cell;
    reaction_48: mIRSp_SHP2 => mIRS + SHP2; k48*mIRSp_SHP2*cell;
    reaction_49: GAB + PIP3 -> mGAB; (k49*GAB*PIP3 - k_49*mGAB)*cell;
    reaction_50: mGAB => mGABp; (kcat50*mGAB*(Rp + alpha50*aSrc)/(Km50 + mGAB))*cell;
    reaction_51: mGABp => mGAB; (V51*mGABp/(Km51 + mGABp))*cell;
    reaction_52: GS + mGABp -> mGABp_GS; (k52*mGABp*GS - k_52*mGABp_GS)*cell;
    reaction_53: PI3K + mGABp -> mGABp_PI3K; (k53*mGABp*PI3K - k_53*mGABp_PI3K)*cell;
    reaction_54: RasGAP + mGABp -> mGABp_RasGAP; (k54*mGABp*RasGAP -
k_54*mGABp_RasGAP)*cell;
    reaction_55: SHP2 + mGABp -> mGABp_SHP2; (k55*mGABp*SHP2 - k_55*mGABp_SHP2)*cell;
    reaction_56: mGABp_SHP2 => SHP2 + mGAB; k56*mGABp_SHP2*cell;
    reaction_57: mGABp_SHP2 => mGABp_pSHP2; (kcat57*mGABp_SHP2*(Rp + aSrc)/(Km57 +
mGABp_SHP2))*cell;
    reaction_58: mGABp_pSHP2 => mGABp_SHP2; (V58*mGABp_pSHP2/(Km58 + mGABp_pSHP2))*cell;
    reaction_59: GS + mGABp_pSHP2 -> mGABp_pSHP2_GS; (k59*mGABp_pSHP2*GS -
k_59*mGABp_pSHP2_GS)*cell;
    reaction_60: => PIP3; k60*(Rp_PI3K + IRp_PI3K + mIRSp_PI3K + mGABp_PI3K +
tRas_PI3K)*cell;
    reaction_61: PIP3 => Null; k61*PIP3*cell;
    reaction_62: dRas => tRas; (kcat62*dRas*(Rp_GS + Rp_pShc_GS + mIRSp_GS + mGABp_GS +
mGABp_pSHP2_GS))/(Km62 + tRas)*cell;
    reaction_63: tRas => dRas; (kcat63*tRas*(bRasGAP + mGABp_RasGAP + Rp_RasGAP +
IRp_RasGAP)/(Km63 + tRas))*cell;
    reaction_64: PI3K + tRas => tRas_PI3K; (k64*tRas*PI3K - k_64*tRas_PI3K)*cell;
    reaction_65: Raf => aRaf; (kcat65*tRas*Raf/(Km65 + Raf))*cell;
    reaction_66: aRaf => aaRaf; (kcat66*aSrc*aRaf/(Km66 + aRaf))*cell;
    reaction_67: aaRaf => Raf; (kcat67*aaRaf*(PKA/(Km67 + aaRaf)) + alpha67*aaRaf*(pAkt +
beta67*ppAkt))*cell;
    reaction_68: Mek => ppMek; (kcat68*aaRaf*Mek/(Km68 + Mek))*cell;
    reaction_69: ppMek => Mek; (V69*ppMek/(Km69 + ppMek))*cell;
    reaction_70: Erk => pErk; (kcat70*Erk*ppMek/(Km70 + Erk + pErk*(Km70/Km71)))*cell;
    reaction_71: pErk => ppErk; (kcat71*pErk*ppMek/(Km71 + pErk + Erk*(Km71/Km70)))*cell;
    reaction_72: ppErk => pErk; (V72*ppErk/(Km72 + ppErk + pErk*(Km72/Km73)))*cell;
    reaction_73: pErk => Erk; (V73*pErk/(Km73 + pErk + ppErk*(Km73/Km72)))*cell;
    reaction_74: PIP3 + PDK1 -> mPDK1; (k74*PDK1*PIP3 - k_74*mPDK1)*cell;
    reaction_75: Akt => pAkt; (kcat75*mPDK1*Akt/(Km75 + Akt))*cell;
    reaction_76: pAkt => Akt; (V76*pAkt/(Km76 + pAkt))*cell;
    reaction_77: mTOR -> amTOR; (kcat77*mTOR*((alpha77*pAkt + ppAkt)/(Km77 + mTOR)) -
k_77*amTOR)*cell;
    reaction_78: pAkt -> ppAkt; (kcat78*amTOR*(pAkt/(Km78 + pAkt)) - k_78*ppAkt)*cell;
    reaction_79: GS -> iGS; (kcat79*ppErk*(GS)/(Km79 + GS)) - k_79*iGS)*cell;
    reaction_80: mGAB -> imGAB; (kcat80*mGAB*(ppErk/(Km80 + mGAB) - k_80*imGAB)*cell;
    reaction_81: mIRS -> imIRS; (kcat81*mIRS*(amTOR/(Km81 + mIRS)) - k_81*imIRS)*cell;
    reaction_82: Rp => Ri; (V_82*Rp/(Km82 + Rp))*cell;
    reaction_83: Ri => Rd; k83*Ri*cell;
    reaction_84: IRp => IRI; (V_84*IRp/(Km84 + IRp))*cell;
    reaction_85: IRI => IRL; k85*IRI*cell;
    reaction_88: mIRSp -> IRSp + PIP3; (k_42*mIRSp - k42*PIP3*IRSp)*cell;
    reaction_89: mIRSp_PI3K -> PIP3 + IRSp_PI3K; (k_42*mIRSp_PI3K -
k42*PIP3*IRSp_PI3K)*cell;
    reaction_90: mIRSp_GS -> PIP3 + IRSp_GS; (k_42*mIRSp_GS - k42*PIP3*IRSp_GS)*cell;
    reaction_91: mIRSp_SHP2 -> PIP3 + IRSp_SHP2; (k_42*mIRSp_SHP2 -
k42*PIP3*IRSp_SHP2)*cell;
    reaction_92: mGABp -> PIP3 + GABp; (k_42*mGABp - k42*PIP3*GABp)*cell;
    reaction_93: mGABp_PI3K -> PIP3 + GABp_PI3K; (k_42*mGABp_PI3K -
k42*PIP3*GABp_PI3K)*cell;
    reaction_94: mGABp_GS -> PIP3 + GABp_GS; (k_42*mGABp_GS - k42*PIP3*GABp_GS)*cell;
    reaction_95: mGABp_RasGAP -> PIP3 + GABp_RasGAP; (k_42*mGABp_RasGAP -
k49*PIP3*GABp_RasGAP)*cell;
    reaction_96: mGABp_SHP2 -> PIP3 + GABp_SHP2; (k_42*mGABp_SHP2 -
k42*PIP3*GABp_SHP2)*cell;
    reaction_97: mGABp_pSHP2 -> PIP3 + GABp_pSHP2; (k_42*mGABp_pSHP2 -
k42*PIP3*GABp_pSHP2)*cell;
    reaction_98: mGABp_pSHP2_GS -> PIP3 + GABp_pSHP2_GS; (k_42*mGABp_pSHP2_GS -
k42*PIP3*GABp_pSHP2_GS)*cell;
    reaction_99: IRSp_PI3K => PI3K + IRS; (V31*IRSp_PI3K/(Km31 + IRSp_PI3K))*cell;
    reaction_100: IRSp_GS => GS + IRS; (V31*IRSp_GS/(Km31 + IRSp_GS))*cell;
    reaction_101: IRSp_SHP2 => IRS + SHP2; k48*IRSp_SHP2*cell;
    reaction_102: mGABp_pSHP2 => SHP2 + mGAB; k56*mGABp_pSHP2*cell;
    reaction_103: mGABp_pSHP2_GS => GS + SHP2 + mGAB; k56*mGABp_pSHP2_GS*cell;
    reaction_104: GABp => GAB; (V51*GABp/(Km51 + GABp))*cell;
    reaction_105: GABp_PI3K => PI3K + GAB; (V51*GABp_PI3K/(Km51 + GABp_PI3K))*cell;
    reaction_106: GABp_GS => GS + GAB; (V51*GABp_GS/(Km51 + GABp_GS))*cell;
    reaction_107: GABp_RasGAP => RasGAP + GAB; (V51*GABp_RasGAP/(Km51 +
GABp_RasGAP))*cell;
    reaction_108: GABp_SHP2 => SHP2 + GAB; k56*GABp_SHP2*cell;
    reaction_109: GABp_pSHP2 => SHP2 + GAB; k56*GABp_pSHP2*cell;
    reaction_110: GABp_pSHP2_GS => GS + SHP2 + GAB; k56*GABp_pSHP2_GS*cell;
    reaction_111: mGABp_RasGAP => RasGAP + mGAB; k111*(mGABp_SHP2 + mGABp_pSHP2 +
mGABp_pSHP2_GS + mIRSp_SHP2)*mGABp_RasGAP*cell;
    reaction_112: Rp_RasGAP => Rp + RasGAP; k111*(mGABp_SHP2 + mGABp_pSHP2 +
mGABp_pSHP2_GS)*Rp_RasGAP*cell;
    reaction_113: IRp_RasGAP => RasGAP + IRp; k111*(mGABp_SHP2 + mGABp_pSHP2 +
mGABp_pSHP2_GS)*IRp_RasGAP*cell;
    reaction_114: Rp_RasGAP => Rp + RasGAP; k111*mIRSp_SHP2*Rp_RasGAP*cell;
    reaction_115: IRp_RasGAP => RasGAP + IRp; k111*mIRSp_SHP2*IRp_RasGAP*cell;
    reaction_117: mGABp -> imGABp; (2*kcat80*mGABp*ppErk/(Km80 + mGABp) -
k_80*imGABp)*cell;
    reaction_118: imGABp => imGAB; k118*imGABp*cell;

    // Species initializations:
    EGF = 1;
    I = 0;
    RE = 0;
    Rd = 0;
    Rp = 0;
    GS = 200;
    Rp_GS = 0;
    Shc = 270;
    Rp_Shc = 0;
    Rp_pShc = 0;
    pShc = 0;
    Rp_pShc_GS = 0;
    PI3K = 200;
    Rp_PI3K = 0;
    RasGAP = 50;
    Rp_RasGAP = 0;
    IRL = 0;
    IRp = 0;
    IRp_PI3K = 0;
    IRp_RasGAP = 0;
    IRS = 300;
    IRp_IRS = 0;
    IRp_IRSp = 0;
    IRSp = 0;
    iSrc = 518;
    mIRS = 0;
    mIRSp = 0;
    mIRSp_GS = 0;
    mIRSp_PI3K = 0;
    SHP2 = 300;
    mIRSp_SHP2 = 0;
    GAB = 225;
    mGAB = 0;
    mGABp = 0;
    mGABp_GS = 0;
    mGABp_PI3K = 0;
    mGABp_SHP2 = 0;
    mGABp_pSHP2 = 0;
    PIP3 = 0;
    dRas = 150;
    Raf = 100;
    aRaf = 0;
    Mek = 200;
    Erk = 400;
    pErk = 0;
    PDK1 = 100;
    Akt = 100;
    pAkt = 0;
    mTOR = 100;
    Null = 0;
    aaRaf = 0;
    PKA = 100;
    pShc_GS = 0;
    ppMek = 0;
    mGABp_pSHP2_GS = 0;
    R = 100;
    ppErk = 0;
    IR = 150;
    mPDK1 = 0;
    tRas = 0;
    tRas_PI3K = 0;
    ppAkt = 0;
    mGABp_RasGAP = 0;
    amTOR = 0;
    iGS = 0;
    imGAB = 0;
    imIRS = 0;
    aSrc = 0;
    Ri = 0;
    Iri = 0;
    iPX = 200;
    aPX = 0;
    aPX_GS = 0;
    IRSp_PI3K = 0;
    IRSp_GS = 0;
    IRSp_SHP2 = 0;
    GABp = 0;
    GABp_PI3K = 0;
    GABp_GS = 0;
    GABp_RasGAP = 0;
    GABp_SHP2 = 0;
    GABp_pSHP2 = 0;
    GABp_pSHP2_GS = 0;
    imGABp = 0;
    bRasGAP = 1e-05;

    // Compartment initializations:
    cell = 1;
    extra = 34;
    EGF_tot has nM;
    k11 has per_nMs;

    // Variable initializations:
    k9 = 0.00666;
    k9 has per_nMs;
    k_1 has per_sec;
    Kd1 = 0.58824;
```



```
Kd1 has nM;
k1 = 0.068;
k1 has per_nMs;
k_2 has per_sec;
Kd2 = 15;
Kd2 has nM;
k2 = 0.033;
k2 has per_nMs;
k_4 has per_sec;
Kd4 = 10;
Kd4 has nM;
k4 = 0.000666;
k4 has per_nMs;
k_5 has per_sec;
Kd5 = 10;
Kd5 has nM;
k5 = 0.0133;
k5 has per_nMs;
k_7 has per_sec;
Kd7 = 400;
Kd7 has nM;
k7 = 0.000666;
k7 has per_nMs;
k_9 has per_sec;
Kd9 = 10;
Kd9 has nM;
k_10 has per_sec;
Kd10 = 400;
Kd10 has nM;
k10 = 0.0004;
k10 has per_nMs;
k_11 has per_sec;
k_12 has per_sec;
Kd12 = 12.45;
Kd12 has nM;
k12 = 0.00933;
k12 has per_nMs;
k_13 has per_sec;
Kd13 = 200;
Kd13 has nM;
k13 = 6.66e-06;
k13 has per_nMs;
k_24 has per_sec;
Kd24 = 0.029412;
Kd24 has nM;
k24 = 0.011322;
k24 has per_nMs;
k_26 has per_sec;
Kd26 = 124.5;
Kd26 has nM;
k26 = 0.00933;
k26 has per_nMs;
k_27 has per_sec;
Kd27 = 2000000;
Kd27 has nM;
k27 = 6.66e-08;
k27 has per_nMs;
k_28 has per_sec;
Kd28 = 3.75;
Kd28 has nM;
k28 = 0.1066;
k28 has per_nMs;
k_30 has per_sec;
Kd30 = 10;
Kd30 has nM;
k30 = 0.0066;
k30 has per_nMs;
k_42 has per_sec;
Kd42 = 10;
Kd42 has nM;
k42 = 0.00666;
k42 has per_nMs;
k_45 has per_sec;
k45 = 0.000666;
k45 has per_nMs;
Kd45 = 100000;
Kd45 has nM;
k_46 has per_sec;
Kd46 = 1;
Kd46 has nM;
k46 = 0.00666;
k46 has per_nMs;
k_47 has per_sec;
Kd47 = 1000;
Kd47 has nM;
k47 = 0.000666;
k47 has per_nMs;
k_49 has per_sec;
Kd49 = 1;
Kd49 has nM;
k49 = 0.000666;
k49 has per_nMs;
k_52 has per_sec;
k52 = 0.002;
k52 has per_nMs;
Kd52 = 1;
Kd52 has nM;
k_53 has per_sec;
Kd53 = 2.5;
Kd53 has nM;
k53 = 0.0133;
k53 has per_nMs;
k_54 has per_sec;
Kd54 = 66666;
Kd54 has nM;
k54 = 1e-05;
k54 has per_nMs;
k_55 has per_sec;
Kd55 = 100;
Kd55 has nM;
k55 = 0.000666;
k55 has per_nMs;
k_59 has per_sec;
Kd59 = 20;
Kd59 has nM;
k59 = 0.01;
k59 has per_nMs;
k_74 has per_sec;
k74 = 0.00666;
k74 has per_nMs;
Kd74 = 100;
Kd74 has nM;
k3 = 0.4;
k3 has per_sec;
k6 = 0.333;
k6 has per_sec;
V8 = 200;
V8 has nM_per_s;
Km8 = 100;
Km8 has nM;
k17 = 0.000185;
k17 has per_sec;
k25 = 1.66;
k25 has per_sec;
k29 = 0.66;
k29 has per_sec;
V31 = 333;
V31 has nM_per_s;
Km31 = 143.3;
Km31 has nM;
kcat40 = 6.6;
kcat40 has per_sec;
alpha40 = 0.00025;
alpha40 has dimensionless;
Km40 = 110;
Km40 has nM;
V41 = 6.66;
V41 has nM_per_s;
Km41 = 50;
Km41 has nM;
kcat43 = 33.3;
kcat43 has per_sec;
alpha43 = 0.05;
alpha43 has dimensionless;
Km43 = 150;
Km43 has nM;
k48 = 0.666;
k48 has per_sec;
kcat50 = 3333;
kcat50 has per_sec;
alpha50 = 0.0001;
alpha50 has dimensionless;
Km50 = 150;
Km50 has nM;
V51 = 333;
V51 has nM_per_s;
Km51 = 130;
Km51 has nM;
k56 = 0.666;
k56 has per_sec;
kcat57 = 0.133;
kcat57 has per_sec;
Km57 = 150;
Km57 has nM;
V58 = 2;
V58 has nM_per_s;
Km58 = 130;
Km58 has nM;
k60 = 4.66;
k60 has per_sec;
k61 = 3.33;
k61 has per_sec;
kcat62 = 5.33;
kcat62 has per_sec;
Km62 = 50;
Km62 has nM;
kcat63 = 20000;
kcat63 has per_sec;
Km63 = 50;
Km63 has nM;
k64 = 0;
k64 has per_nMs;
k_64 = 2.5;
k_64 has per_sec;
kcat65 = 0.1;
kcat65 has per_sec;
Km65 = 400;
Km65 has nM;
kcat66 = 3.33;
kcat66 has per_sec;
Km66 = 10;
Km66 has nM;
kcat67 = 0.666;
kcat67 has per_sec;
Km67 = 10000;
Km67 has nM;
alpha67 = 1e-06;
alpha67 has per_nMs;
beta67 = 2;
beta67 has dimensionless;
kcat68 = 0.133;
kcat68 has per_sec;
Km68 = 50;
Km68 has nM;
V69 = 16.6;
V69 has nM_per_s;
Km69 = 675.299;
Km69 has nM;
kcat70 = 0.333;
kcat70 has per_sec;
Km70 = 500;
Km70 has nM;
Km71 = 500;
Km71 has nM;
kcat71 = 0.666;
kcat71 has per_sec;
V72 = 33.3;
V72 has nM_per_s;
Km72 = 500;
Km72 has nM;
Km73 = 500;
Km73 has nM;
V73 = 23.33;
V73 has nM_per_s;
kcat75 = 4.66;
kcat75 has per_sec;
Km75 = 500;
Km75 has nM;
V76 = 16.66;
V76 has nM_per_s;
Km76 = 1;
Km76 has nM;
kcat77 = 0.666;
kcat77 has per_sec;
alpha77 = 0.5;
alpha77 has dimensionless;
Km77 = 100;
Km77 has nM;
k_77 = 0.666;
k_77 has per_sec;
kcat78 = 0.666;
kcat78 has per_sec;
Km78 = 100;
Km78 has nM;
k_78 = 0.666;
k_78 has per_sec;
kcat79 = 0.0466;
kcat79 has per_sec;
Km79 = 5000;
Km79 has nM;
k_79 = 6.66e-05;
k_79 has per_sec;
kcat80 = 0.04;
kcat80 has per_sec;
Km80 = 700;
Km80 has nM;
k_80 = 6.66e-05;
k_80 has per_sec;
kcat81 = 0.166;
kcat81 has per_sec;
Km81 = 300;
Km81 has nM;
k_81 = 6.66e-05;
k_81 has per_sec;
V_82 = 133;
V_82 has nM_per_s;
Km82 = 50;
Km82 has nM;
k83 = 0.0166;
k83 has per_sec;
V_84 = 333;
V_84 has nM_per_s;
Km84 = 266;
Km84 has nM;
k85 = 0.0166;
k85 has per_sec;
k111 = 0.0133;
k111 has per_nMs;
k118 = 0.001;
k118 has per_sec;

// Other declarations:
var EGF_tot, k11, k_1, k_2, k_4, k_5, k_7, k_9, k_10, k_11, k_12, k_13;
var k_24, k_26, k_27, k_28, k_30, k_42, k_45, k_46, k_47, k_49, k_52, k_53;
var k_54, k_55, k_59, k_74;
const cell, extra, k9, Kd1, k1, Kd2, k2, Kd4, k4, Kd5, k5, Kd7, k7, Kd9;
const Kd10, k10, Kd12, k12, Kd13, k13, Kd24, k24, Kd26, k26, Kd27, k27;
const Kd28, k28, Kd30, k30, Kd42, k42, k45, Kd45, Kd46, k46, Kd47, k47;
```

```
const Kd49, k49, k52, Kd52, Kd53, k53, Kd54, k54, Kd55, k55, Kd59, k59;
const k74, Kd74, k3, k6, V8, Km8, k17, k25, k29, V31, Km31, kcat40, alpha40;
const Km40, V41, Km41, kcat43, alpha43, Km43, k48, kcat50, alpha50, Km50;
const V51, Km51, k56, kcat57, Km57, V58, Km58, k60, k61, kcat62, Km62, kcat63;
const Km63, k64, k_64, kcat65, Km65, kcat66, Km66, kcat67, Km67, alpha67;
const beta67, kcat68, Km68, V69, Km69, kcat70, Km70, Km71, kcat71, V72;
const Km72, Km73, V73, kcat75, Km75, V76, Km76, kcat77, alpha77, Km77, k_77;
const kcat78, Km78, k_78, kcat79, Km79, k_79, kcat80, Km80, k_80, kcat81;
const Km81, k_81, V_82, Km82, k83, V_84, Km84, k85, k111, k118;

// Unit definitions:
unit nM = 1e-9 mole / litre;
unit per_nMs = litre / (1e-9 mole * second);
unit per_sec = 1 / second;
unit nM_per_s = 1e-9 mole / (litre * second);
unit substance = 1e-9 mole;
unit time_unit = second;
unit per_nM = litre / 1e-9 mole;

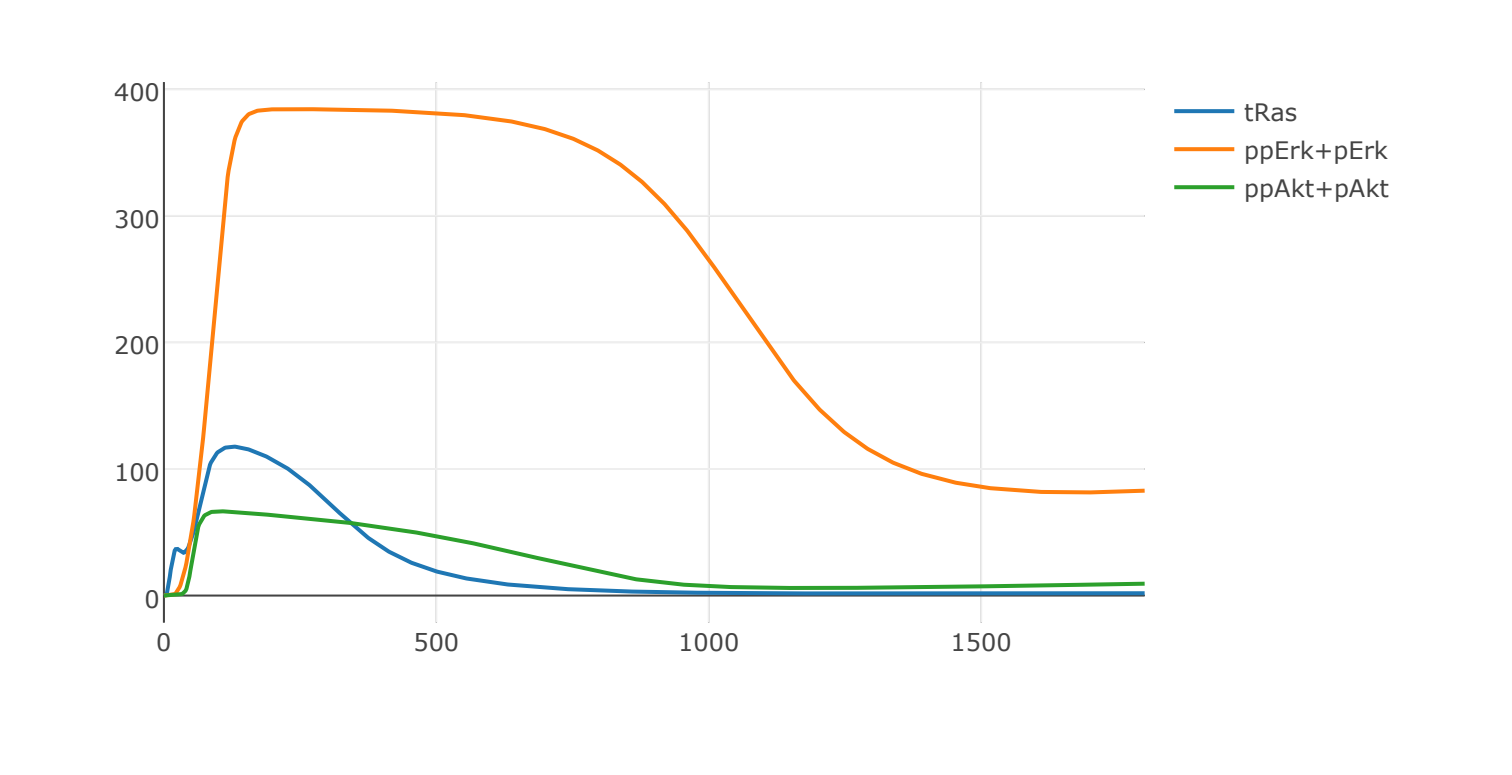
// Display Names:
per_nMs is "per nM per s";
per_sec is "per second";
nM_per_s is "nM per sec";
substance is "nanomole";
time_unit is "seconds";
extra is "extracellular space";
end
// -- End Antimony block

// -- Begin PhraSEdML block converted from main.xml
// Created by libphrasedml v1.0.5
// Models
modell = model "borisov2009"

// Simulations
sim1 = simulate uniform(0, 1800, 1000)

// Tasks
task1 = run sim1 on modell

// Outputs
plot "EGF/Insulin Crosstalk, nM (y) vs sec (x) (Borsiov, 2009)" time vs tRas, ppErk +
pErk, ppAkt + pAkt
// -- End PhraSEdML block
```



[Example 3: p53 Signaling Models](#proctor2008)

Source models: [BIOMD0000000188](https://www.ebi.ac.uk/biomodels-main/BIOMD0000000188), [BIOMD0000000189](https://www.ebi.ac.uk/biomodels-main/BIOMD0000000189)

Reference:

Proctor, C. J., & Gray, D. A. (2008). [Explaining oscillations and variability in the p53-Mdm2 system.](https://bmcsystbiol.biomedcentral.com/articles/10.1186/1752-0509-2-75) BMC systems biology, 2(1), 75.

[Back to top](#topcell)

Example 3: p53 Signaling Models

Source models: [BIOMD0000000188](#), [BIOMD0000000189](#)

Reference:

Proctor, C. J., & Gray, D. A. (2008). [Explaining oscillations and variability in the p53-Mdm2 system](#). BMC systems biology, 2(1), 75.

[Back to top](#)

```
model *p53atm()

// Compartments and Species:
compartment cell;
substanceOnly species Mdm2 in cell, p53 in cell, Mdm2_p53 in cell, Mdm2_mRNA in cell;
substanceOnly species p53_mRNA in cell, ATMA in cell, ATMI in cell, p53_P in cell;
substanceOnly species Mdm2_P in cell, damDNA in cell, $Sink in cell, $Source in cell;
substanceOnly species p53deg in cell, p53syn in cell, mdm2deg in cell, mdm2syn in
cell;
substanceOnly species Mdm2mRNAdeg in cell, Mdm2mRNAAsyn in cell, $totp53 in cell;
substanceOnly species $totMdm2 in cell;

// Assignment Rules:
totp53 := p53 + Mdm2_p53 + p53_P;
totMdm2 := Mdm2 + Mdm2_p53 + Mdm2_P;

// Reactions:
p53mRNASynthesis: $Source => p53_mRNA; ksynp53mRNA*Source;
p53mRNADegradation: p53_mRNA => $Sink; kdegp53mRNA*p53_mRNA;
Mdm2Synthesis: Mdm2_mRNA => Mdm2_mRNA + Mdm2 + mdm2syn; ksynMdm2*Mdm2_mRNA;
Mdm2mRNASynthesis1: p53 => p53 + Mdm2_mRNA + Mdm2mRNAAsyn; ksynMdm2mRNA*p53;
Mdm2mRNASynthesis2: p53_P => p53_P + Mdm2_mRNA + Mdm2mRNAAsyn; ksynMdm2mRNA*p53_P;
Mdm2mRNADegradation: Mdm2_mRNA => $Sink + Mdm2mRNAdeg; kdegMdm2mRNA*Mdm2_mRNA;
Mdm2Degradation: Mdm2 => $Sink + mdm2deg; kdegMdm2*Mdm2*kproteff;
p53Synthesis: p53_mRNA => p53 + p53_mRNA + p53syn; ksynp53*p53_mRNA;
p53Degradation: Mdm2_p53 => Mdm2 + p53deg; kdegp53*Mdm2_p53*kproteff;
P53_Mdm2Binding: p53 + Mdm2 => Mdm2_p53; kbinMdm2p53*p53*Mdm2;
P53_Mdm2Release: Mdm2_p53 => p53 + Mdm2; krelMdm2p53*Mdm2_p53;
DNAdamage: => damDNA; kdam*IR;
DNArepair: damDNA => $Sink; krepair*damDNA;
ATMactivation: damDNA + ATMI => damDNA + ATMA; kactATM*damDNA*ATMI;
p53phosphorylation: p53 + ATMA => p53_P + ATMA; kphosp53*p53*ATMA;
p53dephosphorylation: p53_P => p53; kdephosp53*p53_P;
Mdm2phosphorylation: Mdm2 + ATMA => Mdm2_P + ATMA; kphosMdm2*Mdm2*ATMA;
Mdm2dephosphorylation: Mdm2_P => Mdm2; kdephosMdm2*Mdm2_P;
Mdm2Pdegradation: Mdm2_P => $Sink + mdm2deg; kdegATMMdm2*Mdm2_P;
ATMIinactivation: ATMA => ATMI; kinactATM*ATMA;

// Events:
stressCell: at time >= 3600: IR = 25;
stopStress: at time >= 3660: IR = 0;

// Species initializations:
Mdm2 = 5/cell;
p53 = 5/cell;
Mdm2_p53 = 95/cell;
Mdm2_mRNA = 10/cell;
p53_mRNA = 10/cell;
ATMA = 0;
ATMI = 200/cell;
p53_P = 0;
Mdm2_P = 0;
damDNA = 0;
Sink = 1/cell;
Source = 1/cell;
p53deg = 0;
p53syn = 0;
mdm2deg = 0;
mdm2syn = 0;
Mdm2mRNAdeg = 0;
Mdm2mRNAAsyn = 0;

// Compartment initializations:
cell = 1;

// Variable initializations:
IR = 0;
IR has decagray;
ksynMdm2 = 0.000495;
ksynMdm2 has psec;
kdegMdm2 = 0.000433;
kdegMdm2 has psec;
ksynp53 = 0.006;
ksynp53 has psec;
kdegp53 = 0.000825;
kdegp53 has psec;
kbinMdm2p53 = 0.001155;
kbinMdm2p53 has pmolepsec;
krelMdm2p53 = 1.155e-05;
krelMdm2p53 has psec;
ksynMdm2mRNA = 0.0001;
ksynMdm2mRNA has psec;
kdegMdm2mRNA = 0.0001;
kdegMdm2mRNA has psec;
kactATM = 0.0001;
kactATM has pmolepsec;
kdegATMMdm2 = 0.0004;
kdegATMMdm2 has psec;
kinactATM = 0.0005;
kinactATM has psec;
kphosp53 = 0.0005;
kphosp53 has pmolepsec;
kdephosp53 = 0.5;
kdephosp53 has psec;
kphosMdm2 = 2;
kphosMdm2 has pmolepsec;
kdephosMdm2 = 0.5;
kdephosMdm2 has psec;
kdam = 0.08;
kdam has molepsecpdGy;
krepair = 2e-05;
krepair has psec;
kproteff = 1;
kproteff has dimensionless;
ksynp53mRNA = 0.001;
ksynp53mRNA has psec;
kdegp53mRNA = 0.0001;
kdegp53mRNA has psec;

// Other declarations:
var IR;
const cell, ksynMdm2, kdegMdm2, ksynp53, kdegp53, kbinMdm2p53, krelMdm2p53;
const ksynMdm2mRNA, kdegMdm2mRNA, kactATM, kdegATMMdm2, kinactATM, kphosp53;
const kdegp53, kphosMdm2, kdephosMdm2, kdam, krepair, kproteff, ksynp53mRNA;
const kdegp53mRNA;

// Unit definitions:
unit molepsecpdGy = mole / (second * 10 gray);
unit decagray = 10 gray;
unit molepsec = mole / second;
unit pmolepsec = 1 / (mole * second);
unit psec = 1 / second;

// Display Names:
decagray is "dGy";
molepsec is "molepsec";
pmolepsec is "pmolepsec";
end

model *p53arf()

// Compartments and Species:
compartment cell;
substanceOnly species Mdm2 in cell, p53 in cell, Mdm2_p53 in cell, Mdm2_mRNA in cell;
substanceOnly species ARF in cell, ARF_Mdm2 in cell, damDNA in cell, $Sink in cell;
substanceOnly species $Source in cell, p53deg in cell, p53syn in cell, mdm2deg in
cell;
substanceOnly species mdm2syn in cell, Mdm2mRNAdeg in cell, Mdm2mRNAAsyn in cell;
substanceOnly species totdamDNA in cell, $totp53 in cell, $totMdm2 in cell;

// Assignment Rules:
totp53 := p53 + Mdm2_p53;
totMdm2 := Mdm2 + Mdm2_p53 + ARF_Mdm2;

// Reactions:
Mdm2Synthesis: Mdm2_mRNA => Mdm2_mRNA + Mdm2 + mdm2syn; ksynMdm2*Mdm2_mRNA;
Mdm2mRNASynthesis: p53 => p53 + Mdm2_mRNA + Mdm2mRNAAsyn; ksynMdm2mRNA*p53;
Mdm2mRNADegradation: Mdm2_mRNA => $Sink + Mdm2mRNAdeg; kdegMdm2mRNA*Mdm2_mRNA;
Mdm2Degradation: Mdm2 => $Sink + mdm2deg; kdegMdm2*Mdm2*kproteff;
p53Synthesis: $Source => p53 + p53syn; ksynp53*Source;
p53Degradation: Mdm2_p53 => Mdm2 + p53deg; kdegp53*Mdm2_p53*kproteff;
P53_Mdm2Binding: p53 + Mdm2 => Mdm2_p53; kbinARFMdm2*ARF*Mdm2;
P53_Mdm2Release: Mdm2_p53 => p53 + Mdm2; krelMdm2p53*Mdm2_p53;
DNAdamage: => damDNA + totdamDNA; kdam*IR;
DNArepair: damDNA => $Sink; krepair*damDNA;
ARFactivation: damDNA => damDNA + ARF; kactARF*damDNA;
ARF_Mdm2Binding: ARF + Mdm2 => ARF_Mdm2; kbinARFMdm2*ARF*Mdm2;
ARF_Mdm2Degradation: ARF_Mdm2 => ARF + mdm2deg; kdegARFMdm2*ARF_Mdm2*kproteff;
ARFDegradation: ARF => $Sink; kdegARF*ARF*kproteff;

// Events:
stressCell: at time >= 3600: IR = 25;
stopStress: at time >= 3660: IR = 0;

// Species initializations:
Mdm2 = 5/cell;
p53 = 5/cell;
Mdm2_p53 = 95/cell;
Mdm2_mRNA = 0;
ARF = 0;
ARF_Mdm2 = 0;
damDNA = 0;
```



```
Sink = 1/cell;
Source = 1/cell;
p53deg = 0;
p53syn = 0;
mdm2deg = 0;
mdm2syn = 0;
Mdm2mRNAdeg = 0;
Mdm2mRNAasn = 0;
totdamDNA = 0;

// Compartment initializations:
cell = 1;

// Variable initializations:
IR = 0;
IR has decagray;
ksynMdm2 = 0.000495;
ksynMdm2 has psec;
kdegMdm2 = 0.000433;
kdegMdm2 has psec;
ksynp53 = 0.078;
ksynp53 has psec;
kdegp53 = 0.000825;
kdegp53 has psec;
kbinMdm2p53 = 0.001155;
kbinMdm2p53 has pmolepsec;
krelMdm2p53 = 1.155e-05;
krelMdm2p53 has psec;
ksynMdm2mRNA = 0.0001;
ksynMdm2mRNA has psec;
kdegMdm2mRNA = 0.0001;
kdegMdm2mRNA has psec;
kbinARFMdm2 = 0.01;
kbinARFMdm2 has pmolepsec;
kdegARFMdm2 = 0.001;
kdegARFMdm2 has psec;
kdegARF = 0.0001;
kdegARF has psec;
kactARF = 3.3e-05;
kactARF has psec;
kdam = 0.08;
kdam has molepsecpdGy;
krepare = 2e-05;
krepare has psec;
kproteff = 1;
kproteff has dimensionless;

// Other declarations:
var IR;
const cell, ksynMdm2, kdegMdm2, ksynp53, kdegp53, kbinMdm2p53, krelMdm2p53;
const ksynMdm2mRNA, kdegMdm2mRNA, kbinARFMdm2, kdegARFMdm2, kdegARF, kactARF;
const kdam, krepare, kproteff;

// Unit definitions:
unit molepsecpdGy = mole / (second * 10 gray);
unit decagray = 10 gray;
unit molepsec = mole / second;
unit pmolepsec = 1 / (mole * second);
unit psec = 1 / second;

// Display Names:
decagray is "dGy";
end

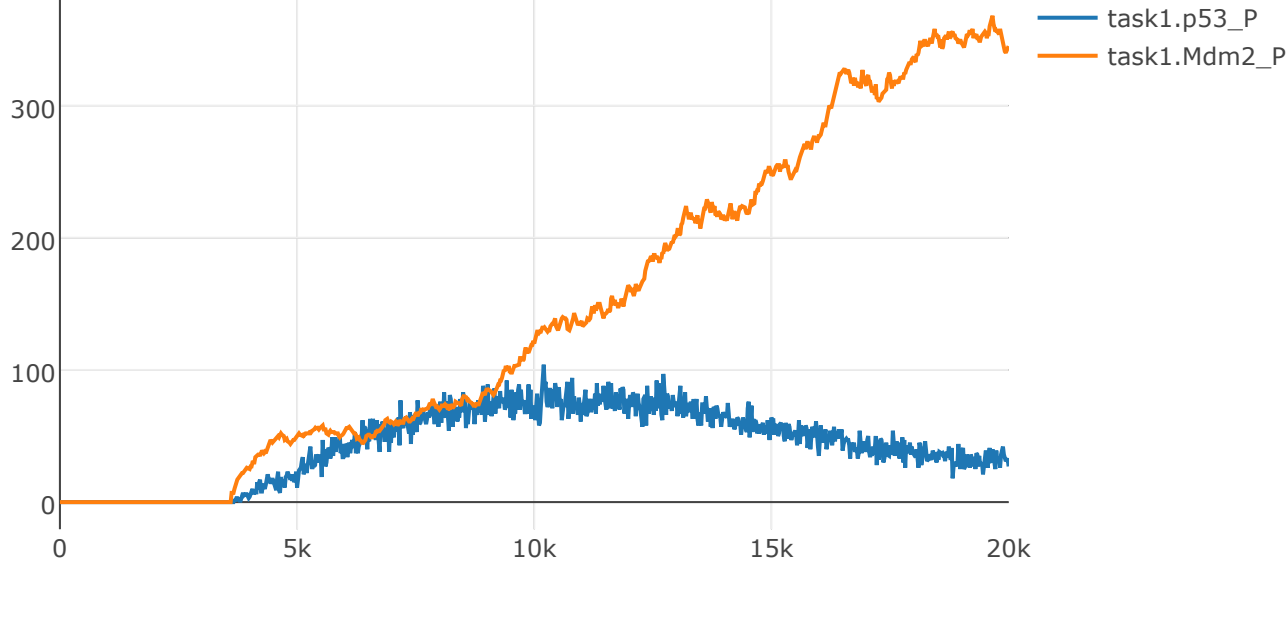
// Models
model1 = model "p53atm"
model2 = model "p53arf"

// Simulation
simulation1 = simulate uniform_stochastic(0, 20000, 1000)

// Tasks
task1 = run simulation1 on model1
task2 = run simulation1 on model2

// Plots
plot "ATM Model: Protein Numbers" task1.time vs task1.p53_P, task1.Mdm2_P
plot "ARF Model: Protein Numbers" task2.time vs task2.p53, task2.ARF
```

ATM Model: Protein Numbers



ARF Model: Protein Numbers

