



// Created by libAntimony v2.9.3

model *vinod2011()

// Compartments and Species:

```
compartment cell_1;
species $Clb2T_1 in cell_1, $Mcm_1 in cell_1, $Clb5T_1 in cell_1, $MBF_1 in cell_1;
species $Cln_1 in cell_1, $Cdc20_1 in cell_1, $Cdh1_1 in cell_1, $Sic1T_1 in cell_1;
species $Swi5_1 in cell_1, $Trim2_1 in cell_1, $Clb2_2 in cell_1, $Sic1_1 in cell_1;
species $Trim5_1 in cell_1, $Clb5_1 in cell_1, $Pds1T_1 in cell_1, $Esp1T_1 in cell_1;
species $PoloT_1 in cell_1, $Polo_1 in cell_1, $Netidep_1 in cell_1, $Netlpp_1 in cell_1;
species $RENT_1 in cell_1, $Cdc14n_1 in cell_1, $RENTp_1 in cell_1, $Cdc14c_1 in cell_1;
species $Tem1_1 in cell_1, $Esp1_1 in cell_1, $Cdc15_1 in cell_1, $MEN_1 in cell_1;
species $Pds1_1 in cell_1, $Esp1b_1 in cell_1, $Net1p_1 in cell_1, $Net1_2 in cell_1;
```

// Assignment Rules:

```
V2_1 := kdc1b2_1 + kdc1b2_2*Cdc20_1 + kdc1b2_3*Cdh1_1;
V6_1 := kdc1b5_1 + kdc1b5_2*Cdc20_1;
Vacdh_1 := kdcdh_1*Cdc14n_1 + kdcdh_2*Cdc14c_1;
Vicdh_1 := kpcdh_1 + kpcdh_2*Clb2_2 + kpcdh_3*Clb5_1;
Vdsic_1 := kdsic_3 + kdsic_1*Clb5_1 + kdsic_2*Clb2_2 + kdsic_4*Cln_1;
Clb2_2 := Clb2T_1 + Clb2nd_1 - Trim2_1;
Sic1_1 := Sic1T_1 - Trim2_1 - Trim5_1;
Clb5_1 := Clb5T_1 - Trim5_1;
Vaswi_1 := kaswi_1*Cdc14n_1 + kaswi_2*Cdc14c_1;
Viswi_1 := kiswi_1 + kiswi_2*Clb2_2 + kiswi_3*Clb5_1;
Vd_1 := (kd_2*PP_1 + kd_1*Cdc14n_1)/(Jnet_1 + Net1T_1 - Netidep_1);
Vp_1 := (kp_3*Clb2_2 + kp_4*MEN_1)/(Jnet_1 + Netidep_1);
Vexp_1 := kexp_1 + kexp_2*MEN_1;
Cdc14c_1 := Cdc14T_1 - Cdc14n_1 - RENT_1;
Esp1_1 := Esp1T_1 - Esp1b_1;
Pds1_1 := Pds1T_1 - Esp1b_1;
Net1_2 := Net1T_1 - Net1p_1 - RENT_1 - Netlpp_1;
PP_1 := PPT_1*((1 + kpp_1*ki_1*Esp1_1)/(1 + ki_1 + Esp1_1));
```

// Rate Rules:

```
Clb2T_1' = ksc1b2_1 + ksc1b2_2*Mcm_1 - V2_1*Clb2T_1;
Mcm_1' = (ksmcm_3 + ksmcm_1*Clb2_2)*(1 - Mcm_1)/(Jmcm_1 + 1 - Mcm_1) -
kdmcm_1*Mcm_1/(Jmcm_1 + Mcm_1);
Clb5T_1' = ksc1b5_2 + ksc1b5_1*MBF_1 - V6_1*Clb5T_1;
MBF_1' = kambf_1*(1 - MBF_1)/(Jmbf_1 + 1 - MBF_1) - (kimbf_1*Clb2_2 +
kimb_3*Clb5_1)*MBF_1/(Jmbf_1 + MBF_1);
Cln_1' = kscln_2 + kscln_1*MBF_1 - kdc1n_1*Cln_1;
Cdc20_1' = ks20_2 + ks20_1*Mcm_1 - (kd20_1 + kd20_2*Cdh1_1)*Cdc20_1;
Cdh1_1' = Vacdh_1*(1 - Cdh1_1)/(Jcdh_1 + 1 - Cdh1_1) - Vicdh_1*Cdh1_1/(Jcdh_1 + Cdh1_1);
Sic1T_1' = kssic_2 + kssic_1*Swi5_1 - Vdsic_1*Sic1T_1;
Swi5_1' = Vaswi_1*(Swi5T_1 - Swi5_1)/(Jswi_1 + Swi5T_1 - Swi5_1) - Viswi_1*Swi5_1/(Jswi_1
+ Swi5_1);
Trim2_1' = kasic2_1*Clb2_2*Sic1_1 - (kdsic2_1 + V2_1 + Vdsic_1)*Trim2_1;
Trim5_1' = kasic5_1*Clb5_1*Sic1_1 - (kdsic5_1 + V6_1 + Vdsic_1)*Trim5_1;
Pds1T_1' = kspds_2 + kspds_1*MBF_1 - (kdpds_1 + kdpds_2*Cdc20_1)*Pds1T_1;
Esp1T_1' = ksesp_1 - kdesp_1*Esp1T_1;
PoloT_1' = kspolo_2 + kspolo_1*Mcm_1 - (kdpolo_1 + kdpolo_2*Cdh1_1)*PoloT_1;
Polo_1' = (kapolo_1 + kapolo_2*Clb2_2)*(PoloT_1 - Polo_1)/(Jpolo_1 + PoloT_1 - Polo_1) -
kipolo_1*Polo_1/(Jpolo_1 + Polo_1) - (kdpolo_1 + kdpolo_2*Cdh1_1)*Polo_1;
Netidep_1' = Vd_1*(Net1T_1 - Netidep_1) - Vp_1*Netidep_1;
Netlpp_1' = kp_1*Polo_1*(Net1T_1 - Netidep_1 - Netlpp_1) - Vd_1*Netlpp_1;
RENT_1' = lanet_1*(Net1T_1 - Netlpp_1 - RENT_1)*Cdc14n_1 - ldnet_1*RENT_1 -
kp_1*Polo_1*RENTp_1;
Cdc14n_1' = ((kp_1*Polo_1*RENTp_1 - lanet_1*(Net1T_1 - Netlpp_1 - RENT_1)*Cdc14n_1) +
ldnet_1*RENT_1 - Vexp_1*Cdc14n_1) + kimp_1*Cdc14c_1;
RENTp_1' = (Vp_1*(RENT_1 - RENTp_1) - Vd_1*RENTp_1) + lanet_1*(Net1T_1 - Netidep_1 -
Netlpp_1 - RENTp_1)*Cdc14n_1 - ldnet_1*RENTp_1 - kp_1*Polo_1*RENTp_1;
Tem1_1' = (katem_1 + katem_2*Polo_1)*(1 - Tem1_1)/(Jtem1_1 + 1 - Tem1_1) - ((kitem_3 +
kitem_2)/(1 + kitem_1*Esp1_1))/(Jtem1_1 + Tem1_1)*Tem1_1;
Cdc15_1' = (kac15_1 + kac15_2*Cdc14c_1)*(1 - Cdc15_1)/(Jcdc15_1 + 1 - Cdc15_1) - (kic15_1
+ kic15_2*Clb2_2)*Cdc15_1/(Jcdc15_1 + Cdc15_1);
MEN_1' = lamen_1*(Tem1_1 - MEN_1)*(Cdc15_1 - MEN_1) - ldmen_1*MEN_1 - ((kitem_3 +
kitem_2)/(1 + kitem_3*Esp1_1))/(Jtem1_1 + Tem1_1)*MEN_1 - ((kic15_1 +
kic15_2*Clb2_2)/(Jcdc15_1 + Cdc15_1))*MEN_1;
Esp1b_1' = lapds_1*Pds1_1*Esp1_1 - (ldpds_1 + kdesp_1 + kdpds_1 +
kdpds_2*Cdc20_1)*Esp1b_1;
```

// Species initializations:

```
Clb2T_1 = 0.999107;
Mcm_1 = 0.996743;
Clb5T_1 = 0.201977;
MBF_1 = 0.001977;
Cln_1 = 0.04079;
Cdc20_1 = 0;
Cdh1_1 = 0;
Sic1T_1 = 0.001683;
Swi5_1 = 0;
Trim2_1 = 0.00145;
Trim5_1 = 0;
Pds1T_1 = 0.601977;
Esp1T_1 = 0.25;
PoloT_1 = 1;
Polo_1 = 1;
Netidep_1 = 0.0119;
Netlpp_1 = 0.0119;
RENT_1 = 0.483;
Cdc14n_1 = 0.00214;
RENTp_1 = 0.014;
Tem1_1 = 1;
Cdc15_1 = 0.933;
MEN_1 = 0;
Esp1b_1 = 0.24857;
Net1p_1 = 0.013;
```

// Compartment initializations:

```
cell_1 = 1;
```

// Variable initializations:

```
ksclb2_1 = 0.015;
ksclb2_2 = 0.005;
ksclb5_2 = 0.002;
ksclb5_1 = 0.01;
kscln_2 = 0.01;
kscln_1 = 0.1;
kdc1n_1 = 0.25;
ks20_2 = 0.001;
ks20_1 = 0.05;
kd20_1 = 0.1;
kd20_2 = 1;
Jcdh_1 = 0.01;
kssic_2 = 0.004;
kssic_1 = 0.2;
kasic2_1 = 40;
kdsic2_1 = 0.1;
kasic5_1 = 10;
kdsic5_1 = 0.1;
Swi5T_1 = 1;
Jswi_1 = 0.1;
ksmcm_3 = 0.01;
ksmcm_1 = 1;
Jmcm_1 = 0.01;
kdmcm_1 = 0.25;
kambf_1 = 0.1;
Jmbf_1 = 0.01;
kimbf_1 = 0.5;
kimbf_3 = 0.5;
kspds_2 = 0.006;
kspds_1 = 0.01;
kdpds_1 = 0.01;
kdpds_2 = 2;
ksesp_1 = 0.001;
kdesp_1 = 0.004;
kspolo_2 = 0.001;
kspolo_1 = 0.05;
kdpolo_1 = 0.05;
kapolo_2 = 0.5;
kapolo_1 = 0;
kapolo_2 = 1;
Jpolo_1 = 0.1;
kipolo_1 = 0.1;
Net1T_1 = 1;
kp_1 = 2;
lanet_1 = 500;
ldnet_1 = 1;
kimp_1 = 1;
katem_1 = 0;
katem_2 = 0.6;
Jtem1_1 = 0.005;
kitem_3 = 0.1;
kitem_2 = 1;
kitem_1 = 20;
kac15_1 = 0.03;
kac15_2 = 0.5;
Jcdc15_1 = 1;
kic15_1 = 0.03;
kic15_2 = 0.2;
lamen_1 = 100;
ldmen_1 = 0.1;
Clb2nd_1 = 0;
lapds_1 = 500;
ldpds_1 = 1;
Cdc14T_1 = 0.5;
PPT_1 = 1;
kpp_1 = 0.1;
ki_1 = 40;
kdc1b2_1 = 0.02;
kdc1b2_2 = 0.1;
kdc1b2_3 = 0.4;
kdc1b5_1 = 0.01;
kdc1b5_2 = 1;
kdsic_3 = 0.04;
kdsic_1 = 2;
kdsic_2 = 2;
kdsic_4 = 1.5;
kdcdh_1 = 0.03;
kdcdh_2 = 0.3;
kpcdh_1 = 0.001;
kpcdh_2 = 0.04;
kpcdh_3 = 0.75;
kaswi_1 = 0.2;
kaswi_2 = 1;
kiswi_1 = 0.01;
kiswi_2 = 0.5;
kiswi_3 = 0.75;
kd_2 = 0.45;
kd_1 = 0.1;
Jnet_1 = 0.05;
kp_3 = 0.2;
kp_4 = 3;
kexp_1 = 0.01;
kexp_2 = 20;
kimbf_2 = 0;
```

// Other declarations:

```
var V2_1, V6_1, Vacdh_1, Vicdh_1, Vdsic_1, Vaswi_1, Viswi_1, Vd_1, Vp_1;
var Vexp_1, lamen_1, PP_1;
const cell_1, ksclb2_1, ksclb2_2, ksclb5_2, ksclb5_1, kscln_2, kscln_1;
const kdc1n_1, ks20_2, ks20_1, kd20_1, kd20_2, Jcdh_1, kssic_2, kssic_1;
const kasic2_1, kdsic2_1, kasic5_1, kdsic5_1, Jswi_1, ksmcm_3;
const ksmcm_1, Jmcm_1, kdmcm_1, kambf_1, Jmbf_1, kimbf_1, kimbf_3, kspds_2;
const kspds_1, kdpds_1, kdpds_2, ksesp_1, kdesp_1, kspolo_1, kspolo_2, kdpolo_1;
const kspolo_2, kapolo_1, kapolo_2, Jpolo_1, kipolo_1, kipolo_2, kdpolo_1, kp_1, lanet_1;
const ldnet_1, katem_1, katem_2, Jtem1_1, kitem_1, kitem_3, kitem_2, kitem_1;
const kac15_1, kac15_2, Jcdc15_1, kic15_1, kic15_2, ldmen_1, Clb2nd_1, lapds_1;
const ldpds_1, Cdc14T_1, PPT_1, kpp_1, ki_1, kdc1b2_1, kdc1b2_2, kdc1b2_3;
const kdc1b5_1, kdc1b5_2, kdsic_3, kdsic_1, kdsic_2, kdsic_4, kdcdh_1, kdcdh_2;
const kpcdh_1, kpcdh_2, kpcdh_3, kaswi_1, kaswi_2, kiswi_1, kiswi_2, kiswi_3;
const kd_2, kd_1, Jnet_1, kp_3, kp_4, kexp_1, kexp_2, kimbf_2;
```

// Unit definitions:

```
unit area_1 = metre^2;
unit length_1 = metre;
unit substance_1 = mole;
unit time_1 = second;
unit volume_1 = litre;
```

// Display Names:

```
area_1 is "area";
length_1 is "length";
substance_1 is "substance";
time_1 is "time";
volume_1 is "volume";
cell_1 is "cell";
Clb2T_1 is "Clb2T";
ksclb2_1 is "ksclb2";
ksclb2_2 is "ksclb2";
Mcm_1 is "Mcm";
V2_1 is "V2";
Clb5T_1 is "Clb5T";
ksclb5_2 is "ksclb5";
ksclb5_1 is "ksclb5";
MBF_1 is "MBF";
V6_1 is "V6";
Cln_1 is "Cln";
kscln_2 is "kscln";
kscln_1 is "kscln";
kdc1n_1 is "kdc1n";
Cdc20_1 is "Cdc20";
ks20_2 is "ks20";
ks20_1 is "ks20";
kd20_1 is "kd20";
kd20_2 is "kd20";
Cdh1_1 is "Cdh1";
Vacdh_1 is "Vacdh";
Jcdh_1 is "Jcdh";
Vicdh_1 is "Vicdh";
Sic1T_1 is "Sic1T";
kssic_2 is "kssic";
kssic_1 is "kssic";
Swi5_1 is "Swi5";
Vdsic_1 is "Vdsic";
Trim2_1 is "Trim2";
kasic2_1 is "kasic2";
Clb2_2 is "Clb2";
Sic1_1 is "Sic1";
kdsic2_1 is "kdsic2";
Trim5_1 is "Trim5";
kasic5_1 is "kasic5";
Clb5_1 is "Clb5";
kdsic5_1 is "kdsic5";
Vaswi_1 is "Vaswi";
Swi5T_1 is "Swi5T";
Jswi_1 is "Jswi";
Viswi_1 is "Viswi";
ksmcm_3 is "ksmcm";
ksmcm_1 is "ksmcm";
Jmcm_1 is "Jmcm";
kdmcm_1 is "kdmcm";
kambf_1 is "kambf";
Jmbf_1 is "Jmbf";
kimbf_1 is "kimbf";
kimbf_3 is "kimbf";
Pds1T_1 is "Pds1T";
kspds_2 is "kspds";
kspds_1 is "kspds";
kdpds_1 is "kdpds";
kdpds_2 is "kdpds";
Esp1T_1 is "Esp1T";
ksesp_1 is "ksesp";
kdesp_1 is "kdesp";
PoloT_1 is "PoloT";
kspolo_2 is "kspolo";
kspolo_1 is "kspolo";
kdpolo_1 is "kdpolo";
kdpolo_2 is "kdpolo";
Polo_1 is "Polo";
kapolo_1 is "kapolo";
kapolo_2 is "kapolo";
Jpolo_1 is "Jpolo";
kipolo_1 is "kipolo";
Netidep_1 is "Netidep";
```

```
Vd_1 is "Vd";
Net1T_1 is "Net1T";
Vp_1 is "Vp";
Net1pp_1 is "Net1pp";
kp_1 is "kp";
RENT_1 is "RENT";
lanet_1 is "lanet";
Cdc14n_1 is "Cdc14n";
ldnet_1 is "ldnet";
RENTp_1 is "RENTp";
Vexp_1 is "Vexp";
kimp_1 is "kimp";
Cdc14c_1 is "Cdc14c";
Tem1_1 is "Tem1";
katem_1 is "katem";
katem_2 is "katem";
Jtem1_1 is "Jtem1";
kitem_3 is "kitem";
kitem_2 is "kitem";
kitem_1 is "kitem";
Esp1_1 is "Esp1";
Cdc15_1 is "Cdc15";
kac15_1 is "kac15";
kac15_2 is "kac15";
Jcdc15_1 is "Jcdc15";
kic15_1 is "kic15";
kic15_2 is "kic15";
MEN_1 is "MEN";
lamen_1 is "lamen";
ldmen_1 is "ldmen";
Clb2nd_1 is "Clb2nd";
Pds1_1 is "Pds1";
Esp1b_1 is "Esp1b";
lapds_1 is "lapds";
ldpds_1 is "ldpds";
Net1p_1 is "Net1p";
Net1_2 is "Net1";
Cdc14T_1 is "Cdc14T";
PP_1 is "pp";
PPT_1 is "ppt";
kpp_1 is "kpp";
ki_1 is "ki";
kdc1b2_1 is "kdc1b2";
kdc1b2_2 is "kdc1b2";
kdc1b2_3 is "kdc1b2";
kdc1b5_1 is "kdc1b5";
kdc1b5_2 is "kdc1b5";
kdsic_3 is "kdsic";
kdsic_1 is "kdsic";
kdsic_2 is "kdsic";
kdsic_4 is "kdsic";
kcdh_1 is "kcdh";
kcdh_2 is "kcdh";
kpcdh_1 is "kpcdh";
kpcdh_2 is "kpcdh";
kpcdh_3 is "kpcdh";
kaswi_1 is "kaswi";
kaswi_2 is "kaswi";
kiswi_1 is "kiswi";
kiswi_2 is "kiswi";
kiswi_3 is "kiswi";
kd_2 is "kd";
kd_1 is "kd";
Jnet_1 is "Jnet";
kp_3 is "kp";
kp_4 is "kp";
kexp_1 is "kexp";
kexp_2 is "kexp";
kimbf_2 is "kimbf";

// SBO terms:
// - Compartment SBO Terms:
cell_1.sboTerm = SBO:0000290;
// - Species SBO Terms:
Clb2T_1.sboTerm = SBO:0000252;
Clb5T_1.sboTerm = SBO:0000252;
Cln_1.sboTerm = SBO:0000252;
Cdc20_1.sboTerm = SBO:0000252;
Cdh1_1.sboTerm = SBO:0000252;
Sic1T_1.sboTerm = SBO:0000252;
Trim2_1.sboTerm = SBO:0000297;
Trim5_1.sboTerm = SBO:0000297;
Swi5_1.sboTerm = SBO:0000252;
Mcm_1.sboTerm = SBO:0000297;
MBF_1.sboTerm = SBO:0000252;
Pds1T_1.sboTerm = SBO:0000252;
Esp1T_1.sboTerm = SBO:0000252;
PoloT_1.sboTerm = SBO:0000252;
Polo_1.sboTerm = SBO:0000252;
Net1dep_1.sboTerm = SBO:0000252;
Net1pp_1.sboTerm = SBO:0000252;
RENT_1.sboTerm = SBO:0000297;
RENTp_1.sboTerm = SBO:0000297;
Cdc14n_1.sboTerm = SBO:0000252;
Tem1_1.sboTerm = SBO:0000252;
Cdc15_1.sboTerm = SBO:0000252;
MEN_1.sboTerm = SBO:0000297;
Clb2_2.sboTerm = SBO:0000252;
Clb5_1.sboTerm = SBO:0000252;
Sic1_1.sboTerm = SBO:0000252;
Pds1_1.sboTerm = SBO:0000252;
Esp1b_1.sboTerm = SBO:0000252;
Esp1_1.sboTerm = SBO:0000252;
Net1p_1.sboTerm = SBO:0000252;
Net1_2.sboTerm = SBO:0000252;
Cdc14c_1.sboTerm = SBO:0000252;
end

// Models
model1 = model "vinod2011"

// Simulations
sim1 = simulate uniform(0, 120, 500)

// Tasks
task1 = run sim1 on model1

// Outputs
plot "Cdc20 Block and Release Experiments (Vinod, 2011)" time vs Clb2T_1, Clb5T_1, Cdc20_1,
Pds1T_1
```

Cdc20 Block and Release Experiments (Vinod, 2011)

