# Examples in SPiM

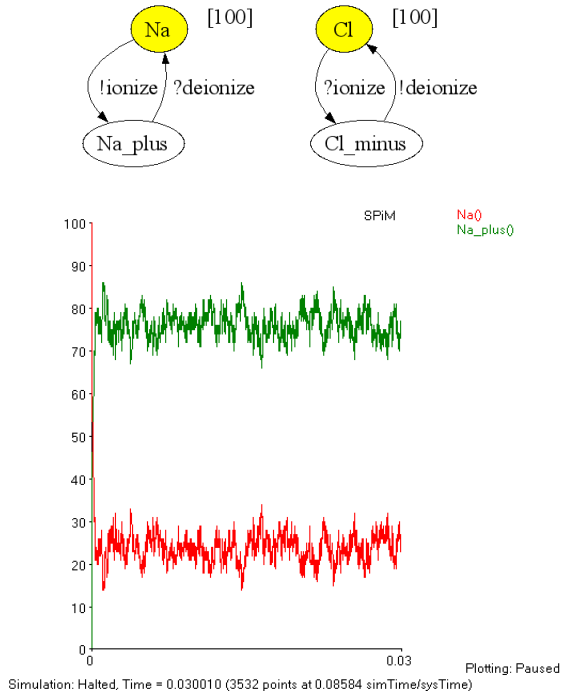
Andrew Phillips

November 29, 2007

## Contents

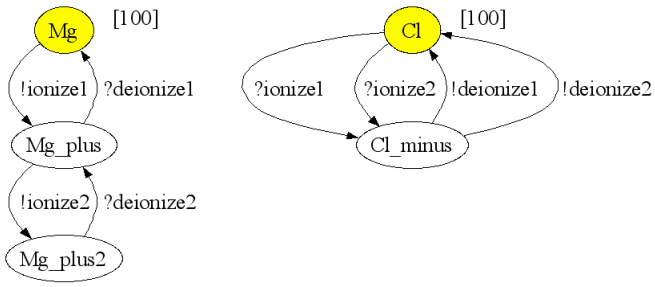
# 1 Chemical Examples

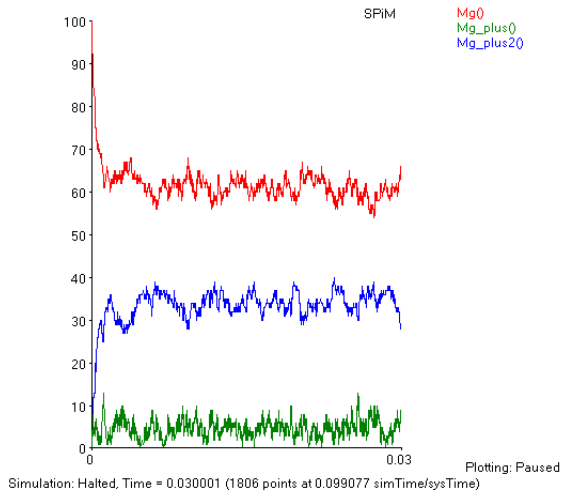
## 1.1 NaCl




```
(* Na + Cl <==> Na+ + Cl- *)
directive sample 0.03
directive plot Na(); Na_plus()
directive graph

new ionize@100.0 : chan
new deionize@10.0 : chan

let Na() = !ionize; Na_plus()
and Na_plus() = ?deionize; Na()

let Cl() = ?ionize; Cl_minus()
and Cl_minus() = !deionize; Cl()

run (100 of Na() | 100 of Cl())
```

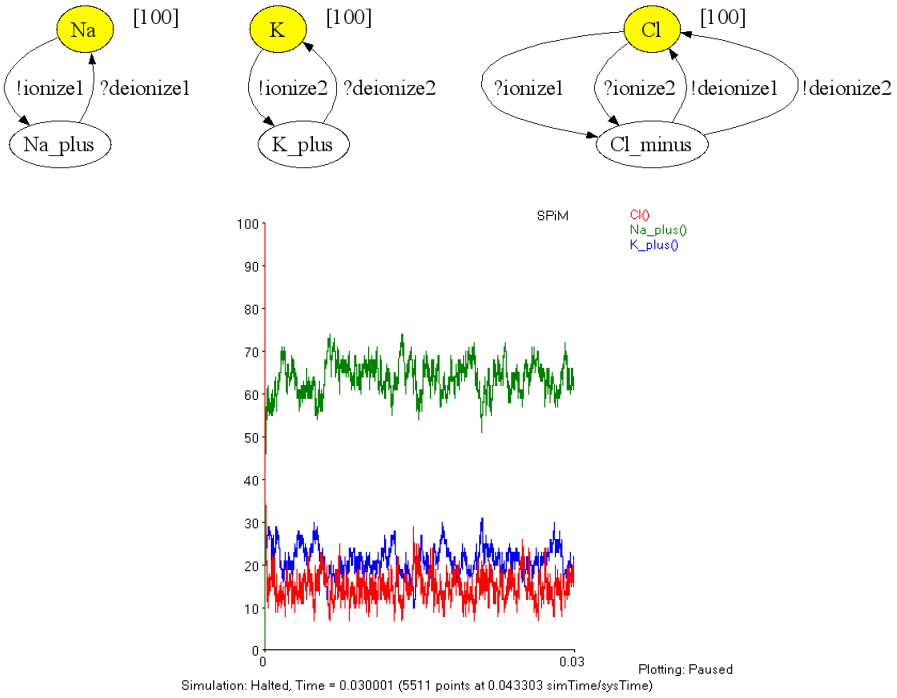
## 1.2 Mg2Cl



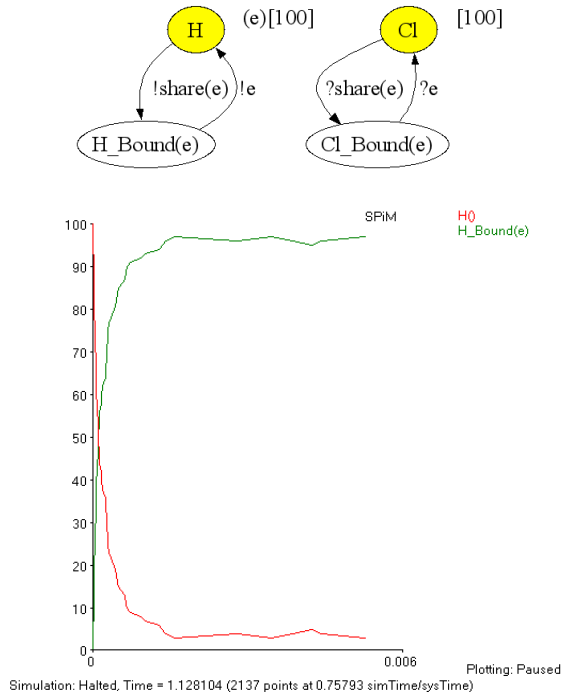

```
(* Mg + 2Cl <==> Mg+2 + 2Cl- *)
directive sample 0.03
directive plot Mg();Mg_plus();Mg_plus2()
directive graph

new ionize1@10.0:chan
new ionize2@100.0:chan
new deionize1@50.0:chan
new deionize2@5.0:chan

let Mg() = !ionize1; Mg_plus()
and Mg_plus() =
  do !ionize2; Mg_plus2()
  or ?deionize1; Mg()
and Mg_plus2() = ?deionize2; Mg_plus()

let Cl() =
  do ?ionize1; Cl_minus()
  or ?ionize2; Cl_minus()
and Cl_minus() =
  do !deionize1; Cl()
  or !deionize2; Cl()

run (100 of Mg() | 100 of Cl())
```

## 1.3 KNa2Cl



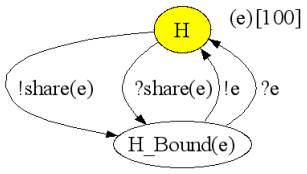

```
(* K + Na + 2Cl <==> K+ + Na+ + 2Cl- *)
directive sample 0.03
directive plot Cl(); Na_plus(); K_plus()
directive graph

new ionize1@100.0:chan
new deionize1@10.0:chan
new ionize2@30.0:chan
new deionize2@20.0:chan

let Na() = !ionize1; Na_plus()
and Na_plus() = ?deionize1; Na()

let K() = !ionize2; K_plus()
and K_plus() = ?deionize2; K()

let Cl() =
  do ?ionize1; Cl_minus()
  or ?ionize2; Cl_minus()
and Cl_minus() =
  do !deionize1; Cl()
  or !deionize2; Cl()

run (100 of Na() | 100 of Cl() | 100 of K())
```

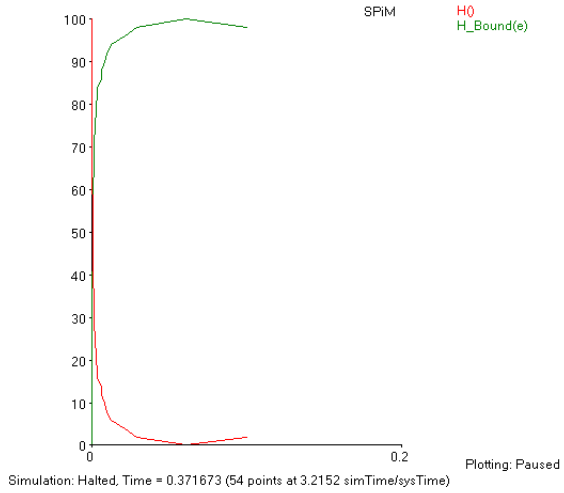
## 1.4 HCl




```
(* H + Cl <==> HCl *)
directive sample 0.006 1000
directive plot H(); H_Bound(e)
directive graph

new share@100.0:chan(chan)

let H() = (
  new e@10.0: chan()
  !share(e); H_Bound(e))
and H_Bound(e:chan) = !e; H()

let Cl() = ?share(e); Cl_Bound(e)
and Cl_Bound(e:chan) = ?e; Cl()

run (100 of H() | 100 of Cl())
```

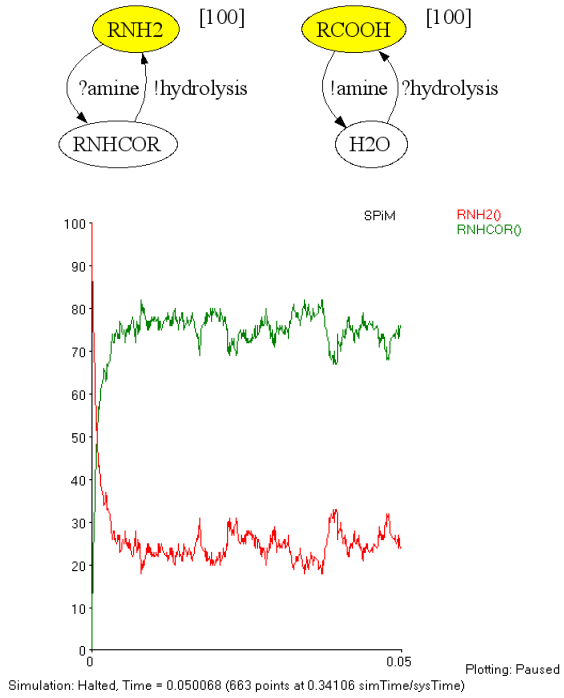
## 1.5 HH



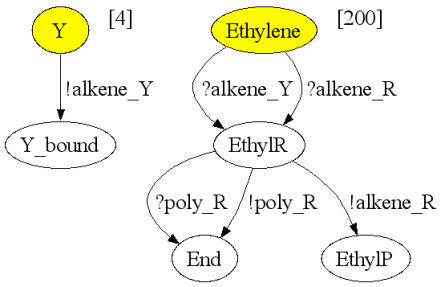

```
(* H + H <==> H2 *)
directive sample 0.2
directive plot H(); H_Bound(e)
directive graph

new share@5.0:chan(chan)

let H() = (
  new e@0.05:chan
  do !share(e); H_Bound(e)
  or ?share(e); H_Bound(e))
and H_Bound(e:chan) =
  do !e; H()
  or ?e; H()

run 100 of H()
```

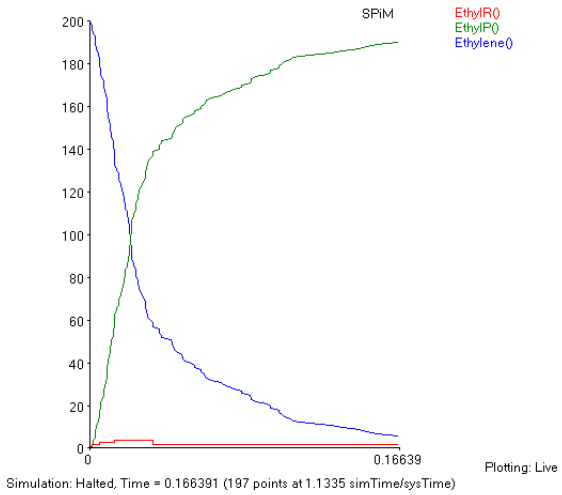
## 1.6 RNHCOR




```
(* RNH2 + RCOOH <==> RNHCOR + H2O *)
directive sample 0.05
directive plot RNH2(); RNHCOR()
directive graph

new amine@10.0:chan
new hydrolysis@1.0:chan

let RNH2() = ?amine; RNHCOR()
and RNHCOR() = !hydrolysis; RNH2()

let RCOOH() = !amine; H2O()
and H2O() = ?hydrolysis; RCOOH()

run (100 of RNH2() | 100 of RCOOH())
```

## 1.7 Ethylene

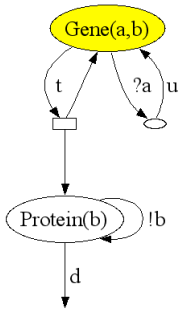


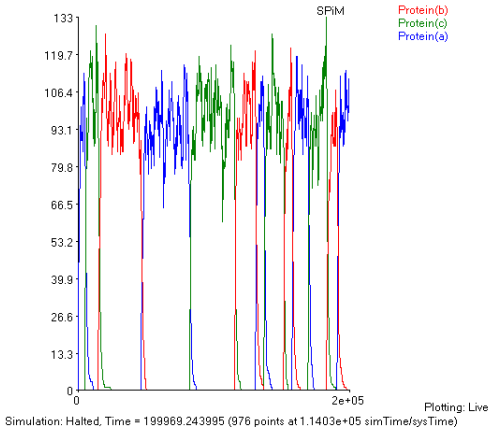




```
(*Y + Ethylene --> Y:EthylR*)
(*:EthylR + Ethylene --> EthylP:EthylR*)
(*:EthylR + :EthylR --> End*)
directive plot EthylR(); EthylP(); Ethylene()
directive graph

new alkene_Y@1.0:chan
new alkene_R@10.0:chan
new poly_R@1.0:chan

let Y() = !alkene_Y; Y_bound()
and Y_bound() = ()

let Ethylene() =
  do ?alkene_Y; EthylR()
  or ?alkene_R; EthylR()
and EthylR() =
  do !alkene_R; EthylP()
  or !poly_R; End()
  or ?poly_R; End()
and EthylP() = ()
and End() = ()

run (200 of Ethylene() | 4 of Y())
```

# 2 Biological Examples

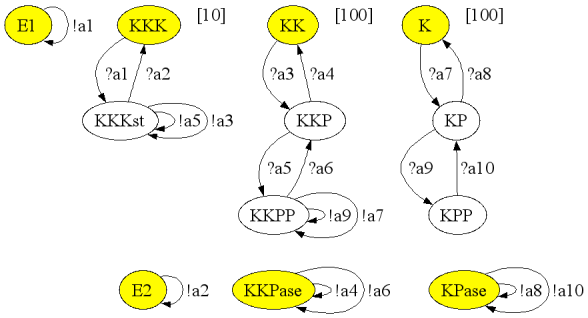
## 2.1 Repressilator

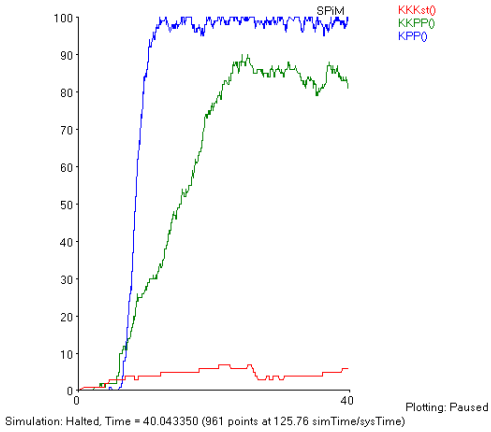



```
(* Repressilator *)
directive sample 200000.0 1000
directive plot Protein(b); Protein(c); Protein(a)
directive graph

val t = 0.1
val d = 0.001
val u = 0.0001
val bind = 1.0
new a@bind:chan
new b@bind:chan
new c@bind:chan

let Gene(a:chan,b:chan) =
  do delay@t; (Protein(b) | Gene(a,b))
  or ?a; delay@u; Gene(a,b)
and Protein(b:chan) =
  do !b; Protein(b)
  or delay@d

run ( Gene(a,b) | Gene(b,c) | Gene(c,a) )
```

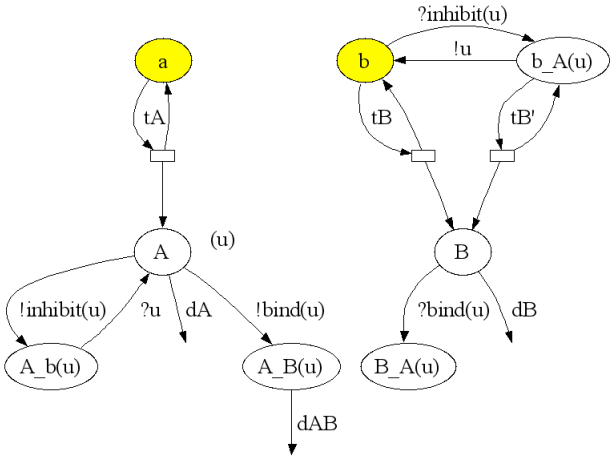
## 2.2 Simplified Mapk Cascade

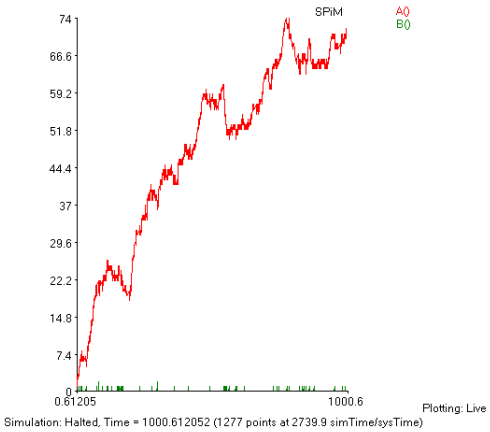


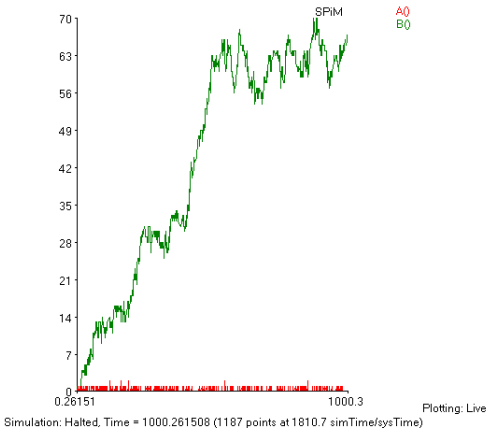




```
(* MAPK cascade *)
directive sample 40.0
directive plot KKKst(); KKPP(); KPP()
directive graph

new a1@0.05:chan
new a2@0.05:chan
new a3@0.05:chan
new a4@0.05:chan
new a5@0.05:chan
new a6@0.05:chan
new a7@0.05:chan
new a8@0.05:chan
new a9@0.05:chan
new a10@0.05:chan

let E1() = !a1; E1()
let E2() = !a2; E2()

let KKK() = ?a1; KKKst()
and KKKst() =
  do ?a2; KKK()
  or !a3; KKKst()
  or !a5; KKKst()

let KK() = ?a3; KKP()
and KKP() =
  do ?a4; KK()
  or ?a5; KKPP()
and KKPP() =
  do ?a6; KKP()
  or !a7; KKPP()
  or !a9; KKPP()

let KKPase() =
  do !a4; KKPase()
  or !a6; KKPase()

let K() = ?a7; KP()
and KP() =
  do ?a8; K()
  or ?a9; KPP()
and KPP() = ?a10; KP()

let KPase() =
  do !a8; KPase()
  or !a10; KPase()

run ( E1() | 10 of KKK() )
run ( 100 of KK() | 100 of K() )
run ( E2() | KKPase() | KPase() )
```

## 2.3 Bistable Gene Network

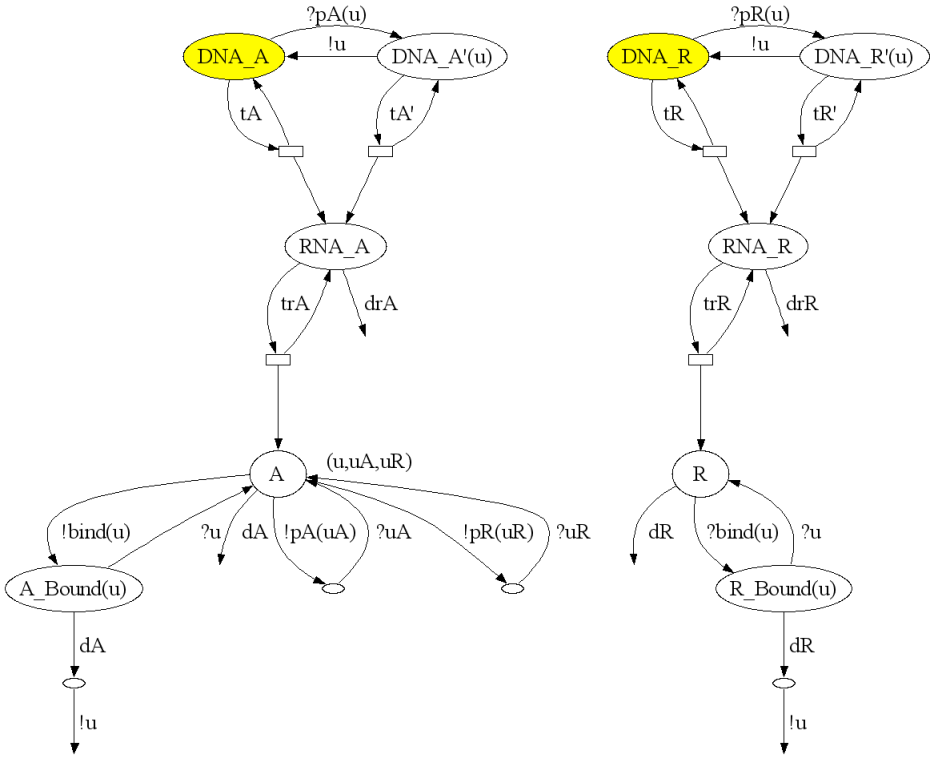


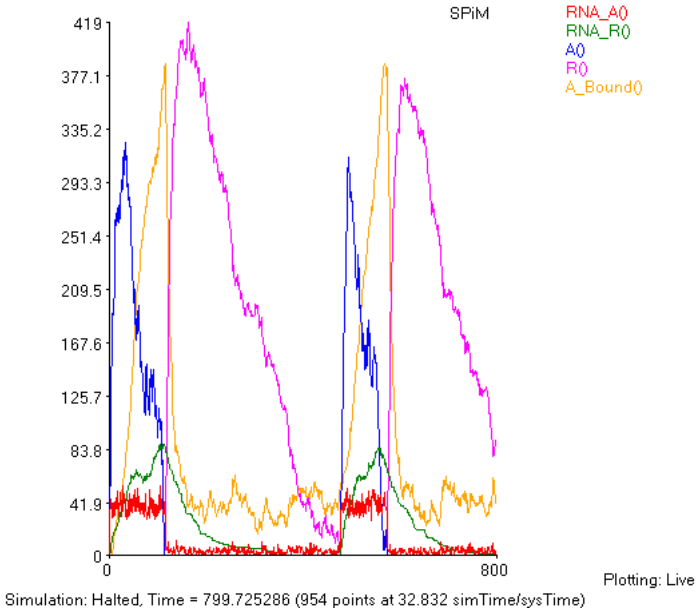




```
directive sample 1000.0
directive plot A(); B()
directive graph

val tA = 0.20
val dA = 0.002 (*0.0085*)
val tB = 0.37
val dB = 0.002 (*0.034*)
new bind@0.72:chan(chan)
val dAB = 0.53
new inhibit@0.19:chan(chan)
val unbind = 0.42
val tB' = 0.027

let a() = delay@tA; ( A() | a() )

and A() = (
  new u@unbind:chan
  do delay@dA
    or !bind(u); A_B(u)
    or !inhibit(u); A_b(u)
)
and A_b(u:chan) = ?u; A()
and A_B(u:chan) = delay@dAB

let b() =
  do delay@tB; ( B() | b() )
  or ?inhibit(u); b_A(u)
and b_A(u:chan) =
  do !u; b()
  or delay@tB'; ( B() | b_A(u) )
and B() =
  do ?bind(u); B_A(u)
  or delay@dB
and B_A(u:chan) = ()

run (a() | b())
```

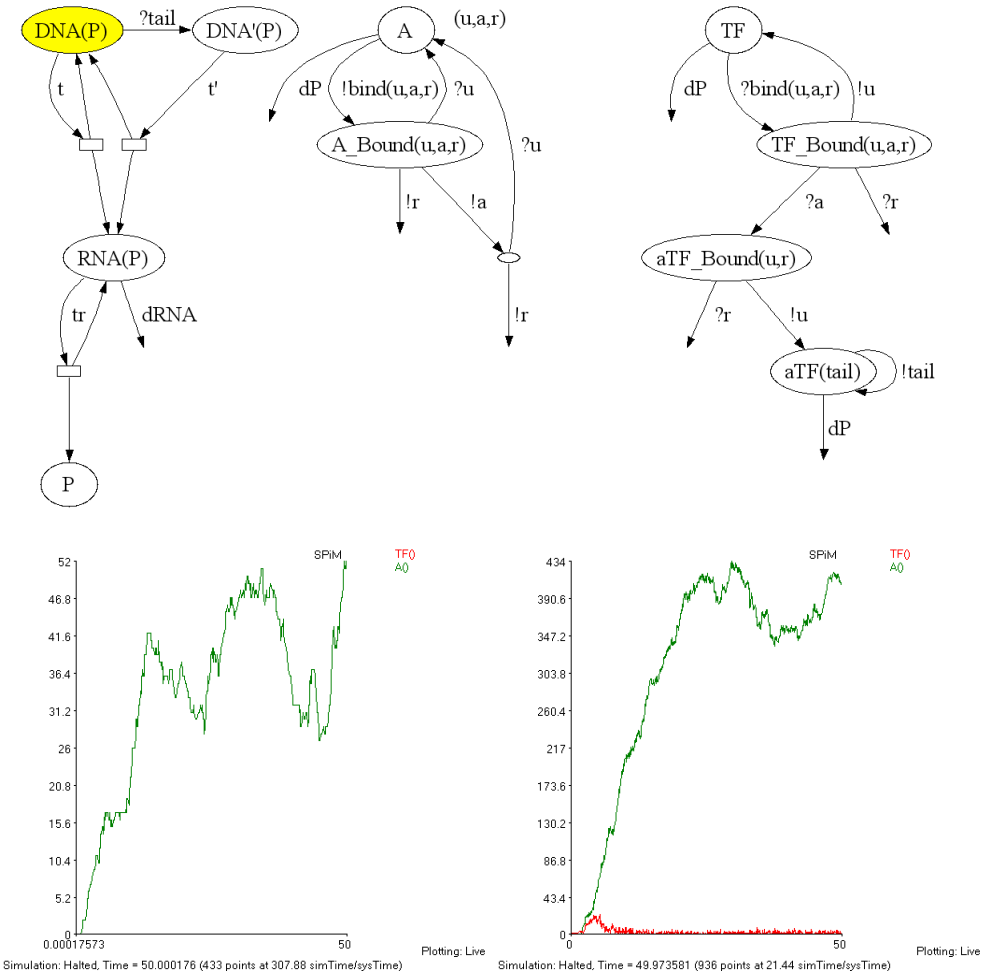
## 2.4 Circadian Clock



```
(* Circadian Clock *)
directive sample 800.0 1000
directive plot RNA_A(); RNA_R(); A(); R(); A_Bound()
directive graph

new bind@100.0:chan(chan)
new pA@10.0:chan(chan)
new pR@10.0:chan(chan)
val drA = 1.0  val dA = 0.1
val drR = 0.02 val dR = 0.01
val tA = 4.0   val tA' = 40.0 val trA = 1.0
val tR = 0.001 val tR' = 2.0  val trR = 0.1

let DNA_A() =
  do delay@tA; (RNA_A() | DNA_A())
  or ?pA(u); DNA_A'(u)
and DNA_A'(u:chan) =
  do delay@tA'; (RNA_A() | DNA_A'(u))
  or !u; DNA_A()
and RNA_A() =
  do delay@trA; (A() | RNA_A())
  or delay@drA
and A() = (
  new u:chan
  new uA@10.0:chan
  new uR@100.0:chan
  do !pA(uA); ?uA; A()
  or !pR(uR); ?uR; A()
  or delay@dA
  or !bind(u); A_Bound(u))
and A_Bound(u:chan) =
    do delay@dA; !u
    or ?u; A()

let DNA_R() =
  do delay@tR; (RNA_R() | DNA_R())
  or ?pR(u); DNA_R'(u)
and DNA_R'(u:chan) =
  do delay@tR'; (RNA_R() | DNA_R'(u))
  or !u; DNA_R()
and RNA_R() =
  do delay@trR; (R() | RNA_R())
  or delay@drR
and R() =
  do delay@dR
  or ?bind(u); R_Bound(u)
and R_Bound(u:chan) =
    do ?u; R()
    or delay@dR; !u

run (DNA_A() | DNA_R())
```

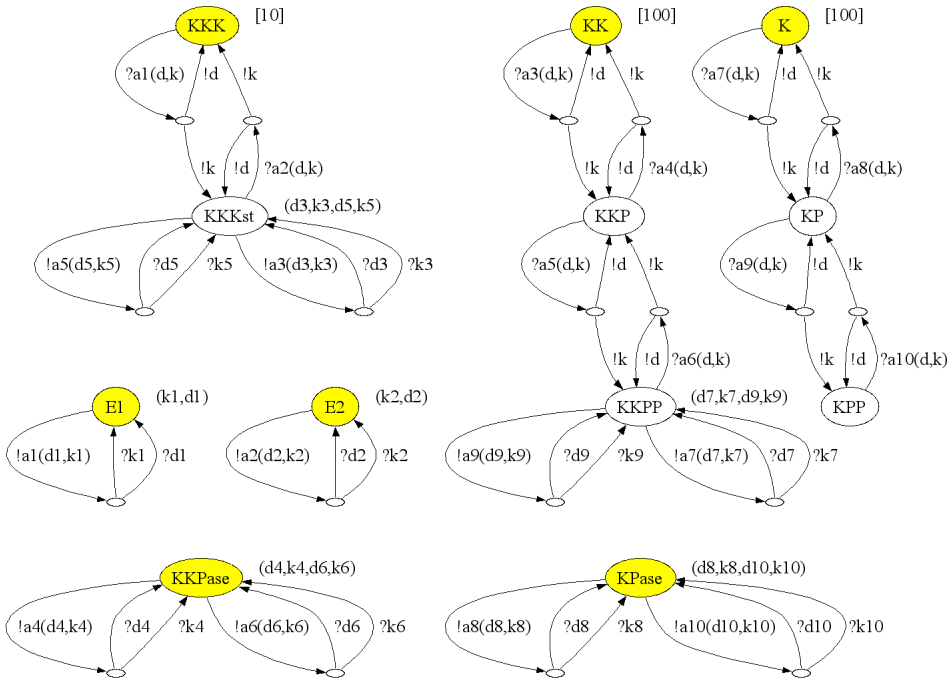
## 2.5 Gene Regulation by Positive Feedback

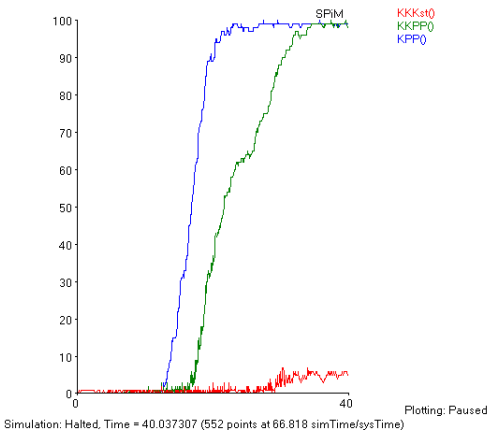



```
(* Gene Regulation by +ve Feedback *)
directive sample 50.0 1000
directive plot TF(); A()
directive graph

val t = 4.0
val t' = 40.0
val tr = 1.0
val dRNA = 1.0
val dP = 0.1
new bind@0.1 : chan(chan,chan,chan)
new tail@100.0: chan

let DNA(P:proc()) =
  do delay@t; (RNA(P) | DNA(P))
  or ?tail; DNA'(P)
and DNA'(P:proc()) =
  delay@t'; (RNA(P) | DNA(P))

and RNA(P:proc()) =
  do delay@tr; (P() | RNA(P))
  or delay@dRNA

let A() = (
  new u@10.0 : chan
  new a@10.0 : chan
  new r@dP: chan
  do delay@dP
  or !bind(u,a,r); A_Bound(u,a,r)
)
and A_Bound(u:chan,a:chan,r:chan) =
  do !r
  or ?u; A()
  or !a;
    do ?u; A()
    or !r

and TF() =
  do delay@dP
  or ?bind(u,a,r); TF_Bound(u,a,r)
```

```
and TF_Bound(u:chan,a:chan,r:chan) =
    do ?r
    or !u; TF()
    or ?a; aTF_Bound(u,r)
and aTF_Bound(u:chan,r:chan) =
      do ?r
           or !u; aTF(tail)

and aTF(tail:chan) =
  do !tail; aTF(tail)
  or delay@dP
run ( DNA(A) | DNA(TF) )
```

## 2.6 Mapk Cascade



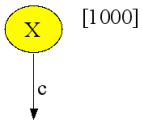



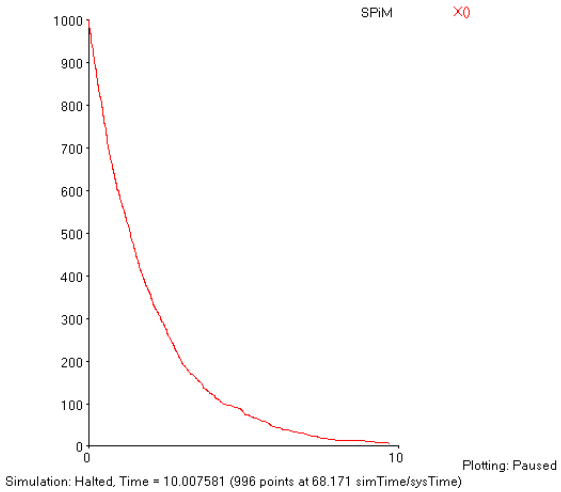

```
(* Huang & Ferrell MAPK cascade *)
directive sample 40.0 1000
directive plot KKKst(); KKPP(); KPP()
directive graph

val ra = 1.0 val rd = 1.0 val rk = 1.0
val rd1=rd val rk1=rk val rd2=rd val rk2=rk
val rd3=rd val rk3=rk val rd4=rd val rk4=rk
val rd5=rd val rk5=rk val rd6=rd val rk6=rk
val rd7=rd val rk7=rk val rd8=rd val rk8=rk
val rd9=rd val rk9=rk val rd10=rd val rk10=rk
type bond = chan(chan,chan)

new a1@ra:bond new a2@ra:bond new a3@ra:bond
new a4@ra:bond new a5@ra:bond new a6@ra:bond
```

```
new a7@ra:bond new a8@ra:bond new a9@ra:bond new a10@ra:bond

let E1() = (
  new k1@rk1:chan new d1@rd1:chan
  !a1(d1,k1); do ?d1; E1() or ?k1; E1()
)
let E2() = (
  new k2@rk2:chan new d2@rd2:chan
  !a2(d2,k2); do ?d2; E2() or ?k2; E2()
)

let KKK() = ?a1(d,k); (do !d; KKK() or !k; KKKst())
and KKKst() = (
  new d3@rd3:chan new k3@rk3:chan
  new d5@rd5:chan new k5@rk5:chan
  do ?a2(d,k); (do !d; KKKst() or !k; KKK())
  or !a3(d3,k3); (do ?d3; KKKst() or ?k3; KKKst())
  or !a5(d5,k5); (do ?d5; KKKst() or ?k5; KKKst())
)
let KK() = ?a3(d,k); (do !d; KK() or !k; KKP())
and KKP() =
  do ?a4(d,k); (do !d; KKP() or !k; KK())
  or ?a5(d,k); (do !d; KKP() or !k; KKPP())
and KKPP() = (
  new d7@rd7:chan new k7@rk7:chan
  new d9@rd9:chan new k9@rk9:chan
  do ?a6(d,k); (do !d; KKPP() or !k; KKP())
  or !a7(d7,k7); (do ?d7; KKPP() or ?k7; KKPP())
  or !a9(d9,k9); (do ?d9; KKPP() or ?k9; KKPP())
)
let K() = ?a7(d,k); (do !d; K() or !k; KP())
and KP() =
  do ?a8(d,k); (do !d; KP() or !k; K())
  or ?a9(d,k); (do !d; KP() or !k; KPP())
and KPP() = ?a10(d,k); (do !d; KPP() or !k; KP())

let KKPase() = (
  new d4@rd4:chan new k4@rk4:chan
  new d6@rd6:chan new k6@rk6:chan
  do !a4(d4,k4); (do ?d4; KKPase() or ?k4; KKPase())
  or !a6(d6,k6); (do ?d6; KKPase() or ?k6; KKPase())
)
let KPase() = (
  new d8@rd8:chan new k8@rk8:chan
  new d10@rd10:chan new k10@rk10:chan
  do !a8(d8,k8); (do ?d8; KPase() or ?k8; KPase())
  or !a10(d10,k10); (do ?d10; KPase() or ?k10; KPase())
)
run ( E1() | 10 of KKK() | 100 of KK() | 100 of K() )
run ( E2() | KKPase() | KPase() )
```

# 3 Gillespie Examples

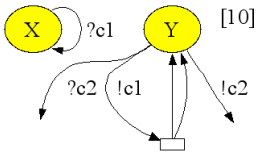
## 3.1 Radioactive Decay

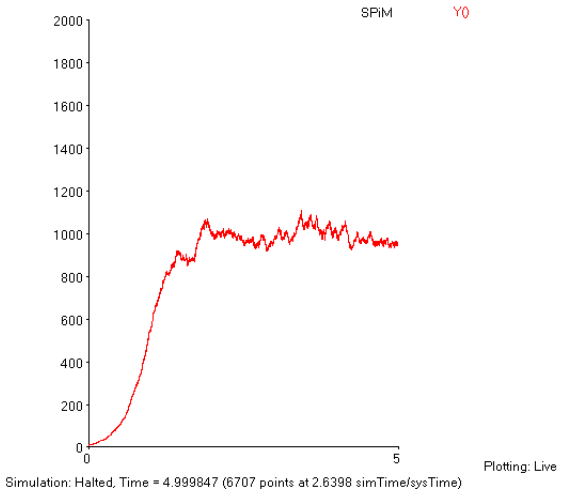



```
directive sample 10.0
directive plot X()
directive graph

val c = 0.5
let X() = delay@c

run 1000 of X() (*1000,5000,10000*)
```

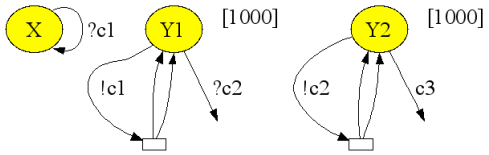
## 3.2 Malek-Mansour and Nicolis

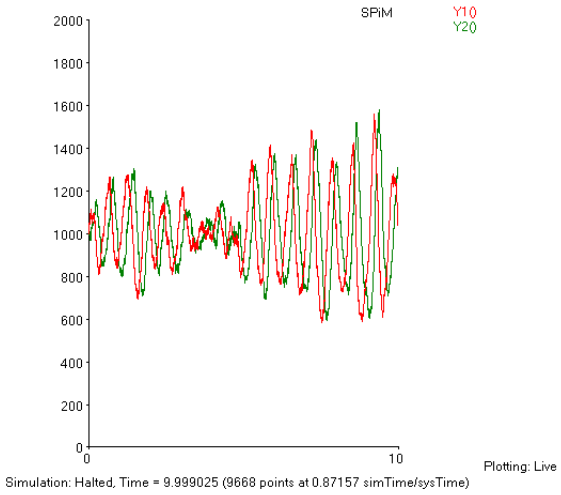


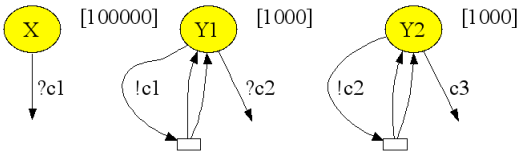

```
directive sample 5.0 10000
directive plot Y()
directive graph

new c1@5.0:chan
new c2@0.0025:chan (*0.0025,0.000625*)

let X() = ?c1; X()
let Y() =
  do !c1; (Y() | Y())
  or !c2; ()
  or ?c2; ()

run (X() | 10 of Y()) (*(10,3000),(40,12000)*)
```

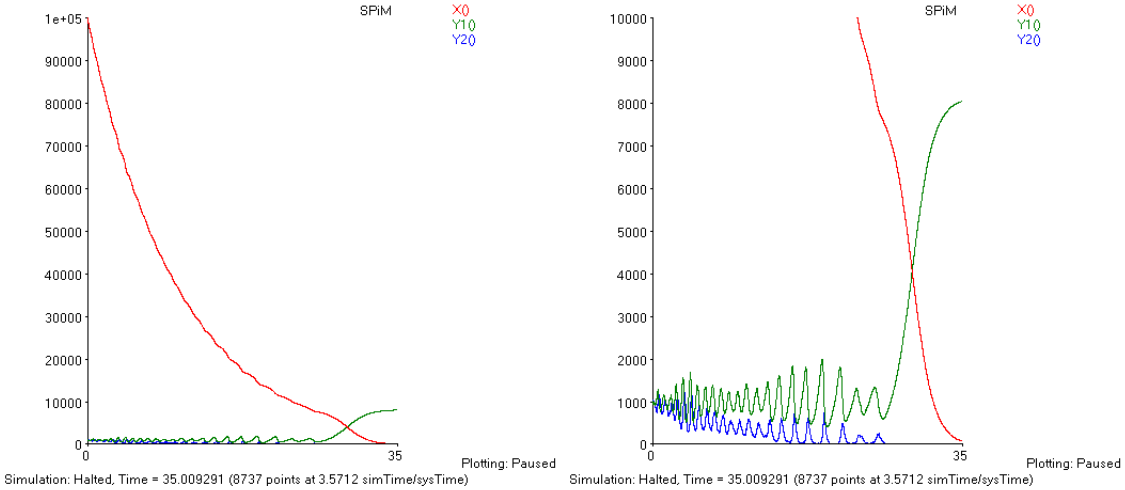
### 3.3 Lotka



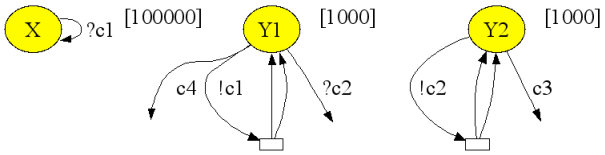



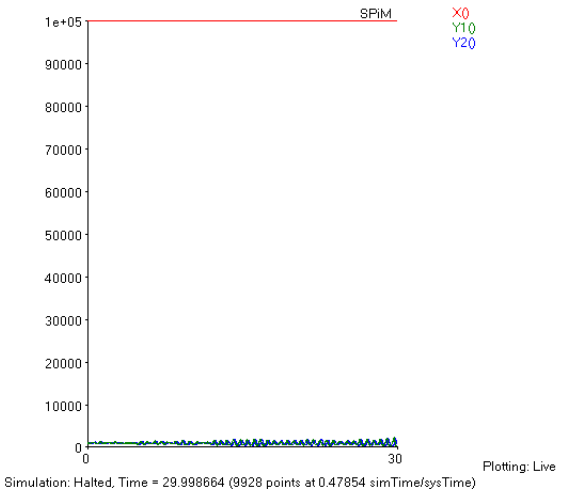

```
directive sample 10.0 10000
directive plot Y1(); Y2()
directive graph

new c1@10.0:chan
new c2@0.01:chan
val c3 = 10.0

let X() = ?c1; X()
let Y1() =
  do !c1; (Y1() | Y1())
  or ?c2
let Y2() =
  do !c2; (Y2() | Y2())
  or delay@c3

run (X() | 1000 of Y1() | 1000 of Y2())
```

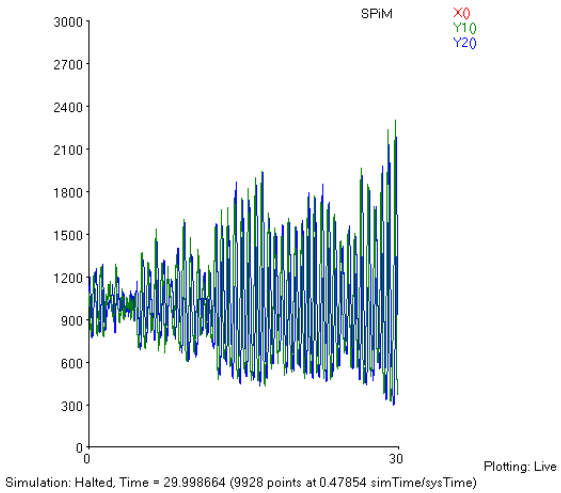


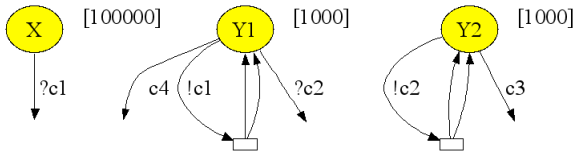

```
directive sample 35.0 10000
directive plot X(); Y1(); Y2()
directive graph

new c1@0.0001:chan
new c2@0.01:chan
val c3 = 10.0

let X() = ?c1
let Y1() =
  do !c1; (Y1() | Y1())
  or ?c2
let Y2() =
  do !c2; (Y2() | Y2())
  or delay@c3

run (100000 of X() | 1000 of Y1() | 1000 of Y2())
```

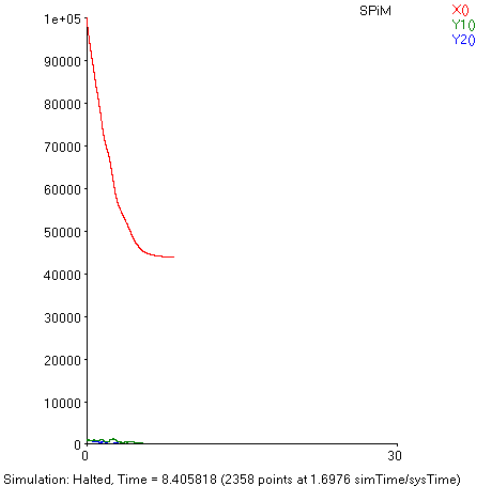


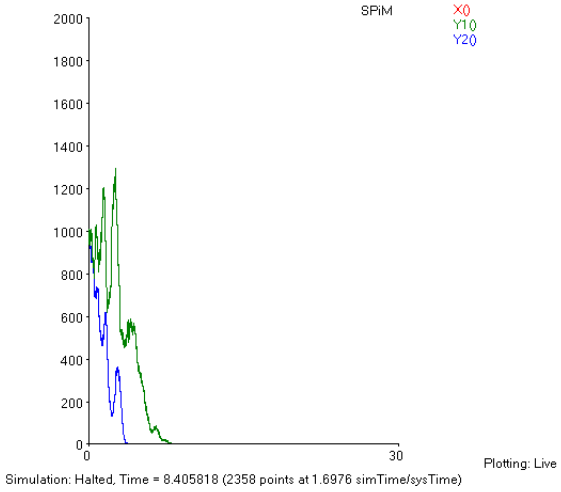




```
directive sample 30.0 10000
directive plot X(); Y1(); Y2()
directive graph

new c1@0.0002:chan
new c2@0.01:chan
val c3 = 10.0
val c4 = 10.0

let X() = ?c1; X()
let Y1() =
  do !c1; (Y1() | Y1())
  or ?c2
  or delay@c4
let Y2() =
  do !c2; (Y2() | Y2())
  or delay@c3

run (100000 of X() | 1000 of Y1() | 1000 of Y2())
```

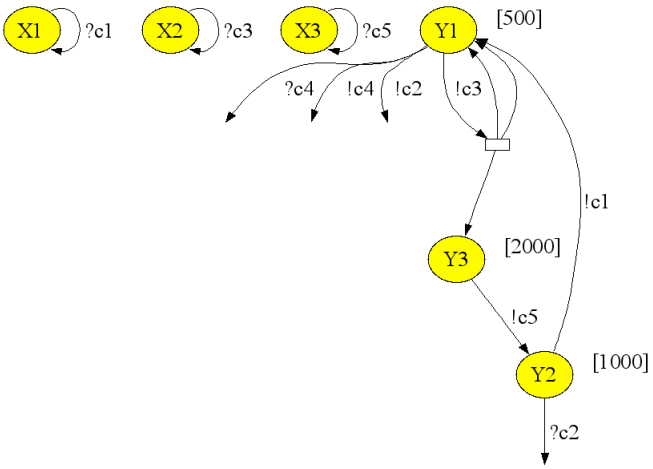


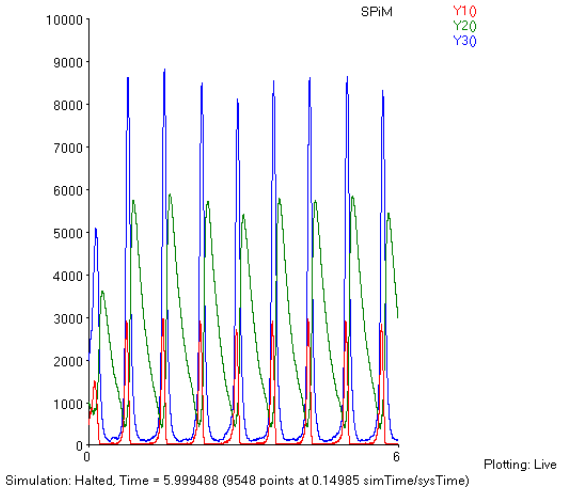




```
directive sample 30.0 10000
directive plot X(); Y1(); Y2()
directive graph

new c1@0.0002:chan
new c2@0.01:chan
val c3 = 10.0
val c4 = 10.0

let X() = ?c1; X()
let Y1() =
  do !c1; (Y1() | Y1())
  or ?c2
  or delay@c4
let Y2() =
  do !c2; (Y2() | Y2())
  or delay@c3

run (100000 of X() | 1000 of Y1() | 1000 of Y2())
```

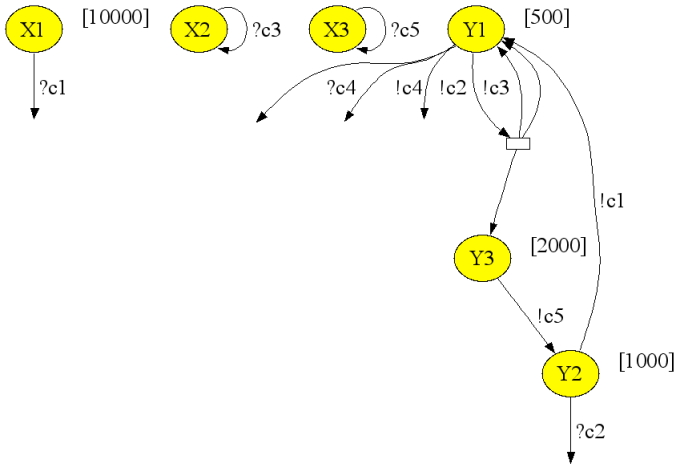
## 3.4 Oregonator

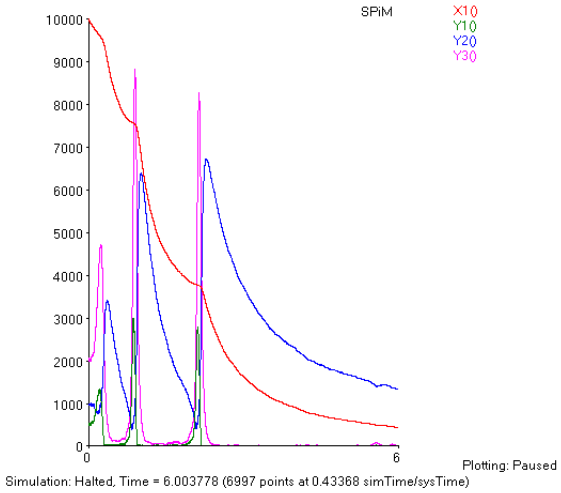



```
directive sample 6.0 10000
directive plot Y1(); Y2(); Y3()
directive graph

new c1@2.0:chan
new c2@0.1:chan
new c3@104.0:chan
new c4@0.008:chan (* 0.016 / 2 *)
new c5@26.0:chan

let X1() = ?c1; X1()
let X2() = ?c3; X2()
let X3() = ?c5; X3()
```

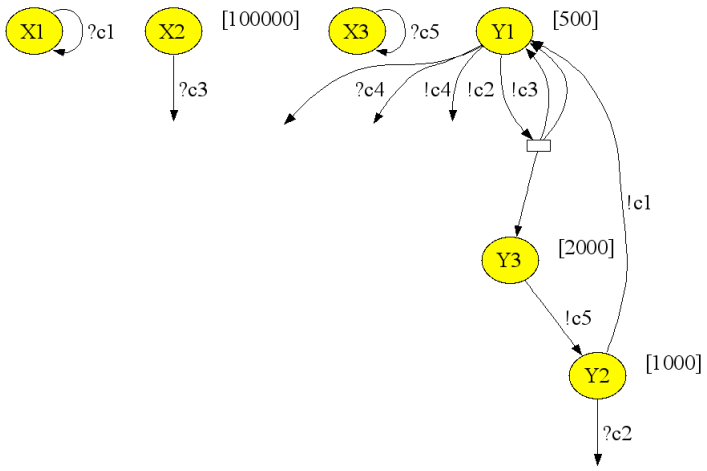
```
let Y1() =
  do !c2
  or !c3; (Y3() | Y1() | Y1())
  or !c4
  or ?c4
and Y2() =
  do !c1; Y1()
  or ?c2
and Y3() = !c5; Y2()

run (X1() | X2() | X3())
run (500 of Y1() | 1000 of Y2() | 2000 of Y3())
```

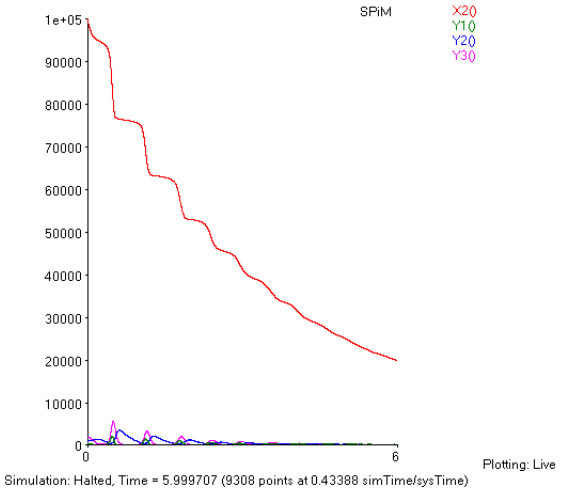


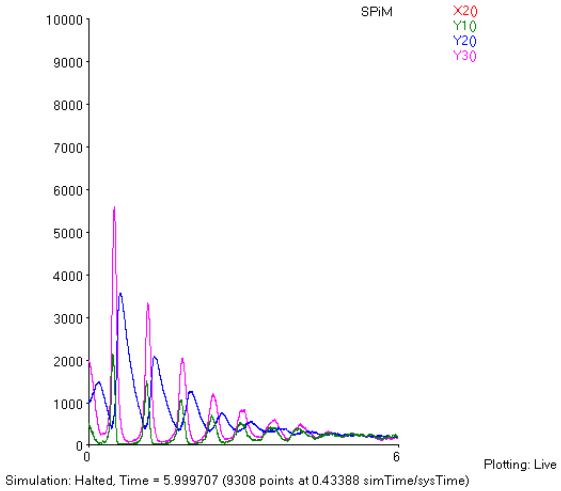




```
directive sample 6.0 10000
directive plot X1(); Y1(); Y2(); Y3()
directive graph

new c1@0.0002:chan
new c2@0.1:chan
new c3@104.0:chan
new c4@0.008:chan (* 0.016 / 2 *)
new c5@26.0:chan

let X1() = ?c1
let X2() = ?c3; X2()
let X3() = ?c5; X3()
```

```
let Y1() =
  do !c2
  or !c3; (Y3() | Y1() | Y1())
  or !c4
  or ?c4
and Y2() =
  do !c1; Y1()
  or ?c2
and Y3() = !c5; Y2()

run (10000 of X1() | X2() | X3())
run (500 of Y1() | 1000 of Y2() | 2000 of Y3())
```




```
directive sample 6.0 10000
directive plot X2(); Y1(); Y2(); Y3()
directive graph

new c1@2.0:chan
new c2@0.1:chan
new c3@0.00104:chan
new c4@0.008:chan (* 0.016 / 2 *)
new c5@26.0:chan

let X1() = ?c1; X1()
let X2() = ?c3
let X3() = ?c5; X3()
```

```
let Y1() =
  do !c2
  or !c3; (Y3() | Y1() | Y1())
  or !c4
  or ?c4
and Y2() =
  do !c1; Y1()
  or ?c2
and Y3() = !c5; Y2()

run (X1() | 100000 of X2() | X3())
run (500 of Y1() | 1000 of Y2() | 2000 of Y3())
```