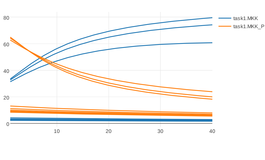
[ ]

[10]
```
// Author Information:
// - Name: kyle medley
// - Email: medleyjduw.edu
// - Organization: uw
%model model1.xml
// -- Begin Antimony block converted from model1.xml
// Created by libAntimony v2.9.3
model *Boris$ID()

  // Compartments and Species:
  compartment compartment_;
  species MKKK in compartment_, MKKK_P in compartment_, MK in compartment_;
  species MK_P in compartment_, MK_PP in compartment_, MAPK in compartment_;
  species MAPK_P in compartment_, MAPK_PP in compartment_;

  // Reactions:
  J0: MKKK => MKKK_P; J0_V1*MKKK/((1 + (MAPK_PP/J0_K1)^J0_n)*(J0_K1 + MKKK));
  J1: MKKK_P => MKKK; J1_V2*MKKK_P/(J1_KK2 + MKKK_P);
  J2: MKK => MK_P; J2_k3*MKKK_P*MKK/(J2_KK3 + MKK);
  J3: MK_P => MKK_PP; J3_k4*MKKK_P*MK_P/(J3_KK4 + MK_P);
  J4: MK_PP => MK_P; J4_V5*MK_PP/(J4_KK5 + MK_PP);
  J5: MK_P => MKK; J5_V6*MK_P/(J5_KK6 + MK_P);
  J6: MAPK => MAPK_P; J6_k7*MK_PP*MAPK/(J6_KK7 + MAPK);
  J7: MAPK_P => MAPK_PP; J7_k8*MK_PP*MAPK_P/(J7_KK8 + MAPK_P);
  J8: MAPK_PP => MAPK_P; J8_V9*MAPK_PP/(J8_KK9 + MAPK_PP);

  // Species initializations:
  MKKK = 90;
  MKKK_P = 10;
  MKK = 100;
  MKK_P = 10;
  MKK_PP = 10;
  MAPK = 100;
  MAPK_P = 10;
  MAPK_PP = 10;

  // Compartment initializations:
  compartment_ = 1;

  // Variable initializations:
  J0_V1 = 2.5;
  J0_K1 = 9;
  J0_n = 1;
  J0_K1 = 10;
  J1_V2 = 0.25;
  J1_KK2 = 8;
  J2_k3 = 0.025;
  J2_KK3 = 15;
  J3_k4 = 0.025;
  J3_KK4 = 15;
  J4_V5 = 0.75;
  J4_KK5 = 15;
  J5_V6 = 0.75;
  J5_KK6 = 15;
  J6_k7 = 0.025;
  J6_KK7 = 15;
  J7_k8 = 0.025;
  J7_KK8 = 15;
  J8_V9 = 0.5;
  J8_KK9 = 15;
  J9_KK10 = 15;

  // Other declarations:
  const compartment_, J0_V1, J0_K1, J0_n, J0_K1, J1_V2, J1_KK2, J2_k3, J2_KK3;
  const J3_k4, J3_KK4, J4_V5, J4_KK5, J5_V6, J5_KK6, J6_k7, J6_KK7, J7_k8;
  const J7_KK8, J8_V9, J8_KK9, J9_V10, J9_KK10;
end
// -- End Antimony block

%tasks simresults.xml --master+True
// -- Begin PhraSEDML block converted from simresults.xml
// Created by libphrasedml v1.0.7
// Models
model1 = model "model1"

// Simulations:
steady1 = simulate steadyState
steady1.algorithm = kisao.282

// Tasks
task0 = run steady1 on model1

// Repeated Tasks
task1 = repeat task2 for model1.J1_KK2 in [1, 5, 10, 50, 60, 70, 80, 90, 100]
task2 = repeat task0 for model1.J4_KK5 in uniform(1, 40, 100)

// Outputs
report "Steady State Values (Boris2D)" task1.J4_KK5, task1.J1_KK2, task1.MKK, task1.MKK_P
plot "Steady State Scan (Boris 2D)" task1.J4_KK5 vs task1.MKK, task1.MKK_P
// -- End PhraSEDML block
```

```
------------------------------------------------------------------------------
   task1.J4_KK5  task1.J1_KK2  task1.MKK  task1.MKK_P
0          1.00          1.0  33.465751    64.942807
1          1.39          1.0  34.607356    63.784046
2          1.78          1.0  35.750746    62.624230
3          2.17          1.0  36.892899    61.466340
4          2.56          1.0  38.038022    60.313320
```



[ ]