Computer code used in

Mathematical modeling of intratumoral immunotherapy yields strategies to improve the treatment outcomes

```
function out = model

import com.comsol.model.*
import com.comsol.model.util.*

model = ModelUtil.create('Model');

model.modelPath('');

model.component.create('comp1', true);

model.component('comp1').geom.create('geom1', 1);

model.component('comp1').mesh.create('mesh1');

model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study.create('std1');
model.study('std1').create('time', 'Transient');
model.study('std1').feature('time').activate('g', true);

model.component('comp1').geom('geom1').run;

model.component('comp1').physics('g').prop('Units').set('CustomDependentVariableUn
it', '1');
model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity
', 'massconcentration');
model.component('comp1').physics('g').prop('Units').setIndex('CustomSourceTermUnit
', 'mol/m/s', 0, 0);
model.component('comp1').physics('g').field('dimensionless').field('Ic_f');
model.component('comp1').physics('g').field('dimensionless').component(1, 'Ic_f');
model.component('comp1').physics('g').feature('gfeq1').setIndex('Ga', '(-
D_Ic_f*Ic_fx*x^2)+(uf*Ic_f*x^2)', 0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('f', 'R_Ic_f*x^2',
0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics.create('g2', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g2', true);

model.component('comp1').physics('g2').prop('Units').set('CustomDependentVariableU
nit', '1');
model.component('comp1').physics('g2').prop('Units').set('DependentVariableQuantit
y', 'massconcentration');
model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity
', 'concentration');
model.component('comp1').physics('g2').prop('Units').set('CustomDependentVariableU
nit', 'kg/m^3');
model.component('comp1').physics('g2').prop('Units').set('DependentVariableQuantit
y', 'concentration');
model.component('comp1').physics('g2').field('dimensionless').field('Ic_b');
model.component('comp1').physics('g2').field('dimensionless').component(1,
'Ic_b');
```

```
model.component('comp1').physics('g2').prop('Units').setIndex('CustomSourceTermUnit', 'mol/m/s', 0, 0);
model.component('comp1').physics('g2').feature('gfeq1').setIndex('Ga', 'xt*Ic_b*x^2', 0);
model.component('comp1').physics('g2').feature('gfeq1').setIndex('f', 'R_Ic_b*x^2', 0);
model.component('comp1').physics('g2').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics.create('g3', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g3', true);

model.component('comp1').physics('g').tag('Ic_f');
model.component('comp1').physics('g2').tag('Ic_b');
model.component('comp1').physics('g3').field('dimensionless').component(1, 'c_c');
model.component('comp1').physics('g3').field('dimensionless').field('c_c');
model.component('comp1').physics('g3').tag('c_c');
model.component('comp1').physics('c_c').prop('Units').set('DependentVariableQuantity', 'massconcentration');
model.component('comp1').physics('c_c').prop('Units').setIndex('CustomSourceTermUnit', 'kg/m/s', 0, 0);
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u2'});

model.study('std1').feature('time').activate('g', true);

model.component('comp1').physics('g').field('dimensionless').field('u2');
model.component('comp1').physics('g').prop('Units').setIndex('CustomSourceTermUnit', 'kg/m/s', 0, 0);
model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity', 'massconcentration');
model.component('comp1').physics('g').field('dimensionless').field('IAPC');
model.component('comp1').physics('g').field('dimensionless').component(1, 'IAPC');
model.component('comp1').physics('g').tag('IAPC');
model.component('comp1').physics('IAPC').feature('gfeq1').setIndex('Ga', '-D_IAPC*IAPCx*x^2+xt*IAPC*x^2', 0);
model.component('comp1').physics('IAPC').feature('gfeq1').setIndex('f', 'R_IAPC*x^2', 0);
model.component('comp1').physics('IAPC').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics.create('g2', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g2', true);

model.component('comp1').physics('g2').identifier('g2');
model.component('comp1').physics('g2').field('dimensionless').field('APC');
model.component('comp1').physics('g2').field('dimensionless').component(1, 'APC');
model.component('comp1').physics('g2').tag('APC');
model.component('comp1').physics('APC').prop('Units').setIndex('CustomSourceTermUnit', 'kg/m/s', 0, 0);
model.component('comp1').physics('APC').prop('Units').set('DependentVariableQuantity', 'massconcentration');
model.component('comp1').physics('APC').feature('gfeq1').setIndex('Ga', '-D_APC*APCx*x^2+xt*APC*x^2', 0);
model.component('comp1').physics('APC').feature('gfeq1').setIndex('f', 'R_APC*x^2', 0);
model.component('comp1').physics('APC').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g', true);
```

```
model.component('comp1').physics('g').tag('Th_E');
model.component('comp1').physics('Th_E').field('dimensionless').field('Th_E');
model.component('comp1').physics('Th_E').field('dimensionless').component(1,
'Th_E');
model.component('comp1').physics('Th_E').prop('Units').set('DependentVariableQuant
ity', 'massconcentration');
model.component('comp1').physics('Th_E').prop('Units').setIndex('CustomSourceTermU
nit', 'kg/m/s', 0, 0);
model.component('comp1').physics('Th_E').feature('gfeq1').setIndex('Ga', '-
D_Th_E*Th_Ex*x^2+xt*Th_E*x^2', 0);
model.component('comp1').physics('Th_E').feature('gfeq1').setIndex('f',
'R_Th_E*x^2', 0);
model.component('comp1').physics('Th_E').feature('gfeq1').setIndex('da', 'x^2',
0);
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g', true);

model.component('comp1').physics('g').feature('gfeq1').setIndex('Ga', '-
D_T_E*T_Ex*x^2+xt*T_E*x^2', 0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('f', 'R_T_E*x^2',
0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('da', 'R_T_E*x^2',
0);
model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity
', 'massconcentration');
model.component('comp1').physics('g').prop('Units').setIndex('CustomSourceTermUnit
', 'kg/m/s', 0, 0);
model.component('comp1').physics('g').field('dimensionless').component(1, 'T_E');
model.component('comp1').physics('g').field('dimensionless').field('T_E');
model.component('comp1').physics('g').tag('T_E');
model.component('comp1').physics.create('g2', 'GeneralFormPDE', {'u2'});

model.study('std1').feature('time').activate('g2', true);

model.component('comp1').physics('g2').field('dimensionless').field('u2');
model.component('comp1').physics('g2').identifier('g2');
model.component('comp1').physics('g2').feature('gfeq1').setIndex('Ga', '(-
D_Ag*Agx*x^2)+(uf*Ag*x^2)', 0);
model.component('comp1').physics('g2').feature('gfeq1').setIndex('f', 'R_Ag*x^2',
0);
model.component('comp1').physics('g2').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g', true);

model.component('comp1').physics('g').field('dimensionless').component(1, 'T');
model.component('comp1').physics('g').field('dimensionless').field('T');
model.component('comp1').physics('g').tag('T');
model.component('comp1').physics('g2').field('dimensionless').component(1, 'Ag');
model.component('comp1').physics('g2').field('dimensionless').field('Ag');
model.component('comp1').physics('g2').prop('Units').set('DependentVariableQuantit
y', 'massconcentration');
model.component('comp1').physics('g2').prop('Units').setIndex('CustomSourceTermUni
t', 'kg/m/s', 0, 0);
model.component('comp1').physics('T').prop('Units').set('DependentVariableQuantity
', 'massconcentration');
model.component('comp1').physics('T').prop('Units').setIndex('CustomSourceTermUnit
', 'kg/m/s', 0, 0);
```

```
model.component('comp1').physics('T').feature('gfeq1').setIndex('Ga', '-
D_T*Tx*x^2+xt*T*x^2', 0);
model.component('comp1').physics('T').feature('gfeq1').setIndex('f', 'R_T*x^2',
0);
model.component('comp1').physics('T').feature('gfeq1').setIndex('da', 'v', 0);
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g', true);

model.component('comp1').physics('g').prop('Units').setIndex('CustomSourceTermUnit
', 'm^2/s', 0, 0);
model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity
', 'pressure');
model.component('comp1').physics('g').field('dimensionless').field('p');
model.component('comp1').physics('g').field('dimensionless').component(1, 'p');
model.component('comp1').physics('g').feature('gfeq1').setIndex('Ga', '-
kh*px*x^2+xt*x^2', 0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('f', '(lp*Sv*(Pv-
p)-lplSv*(p-pl))*x^2', 0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('da', 0, 0);
model.component('comp1').physics.create('g3', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g3', true);

model.component('comp1').physics('g3').feature('gfeq1').setIndex('Ga', '(-
Dox*coxx*x^2)+(uf*cox*x^2)', 0);
model.component('comp1').physics('g3').feature('gfeq1').setIndex('f', '((-
((Aox*cox)/(kox+cox)))*(T/T_inT)+Perox*Sv*(Ciox-cox))*x^2', 0);
model.component('comp1').physics('g3').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics('g3').field('dimensionless').component(1, 'cox');
model.component('comp1').physics('g3').field('dimensionless').field('cox');
model.component('comp1').physics('g3').tag('cox');
model.component('comp1').physics.create('ge', 'GlobalEquations', 'geom1');

model.study('std1').feature('time').activate('ge', true);

model.component('comp1').physics('ge').feature('ge1').setIndex('name', 'Ic_blood',
0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('equation',
'd(Ic_blood,t)-R_Ic_blood', 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('initialValueU',
'Ic_blood_in', 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('initialValueUt',
0, 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('description', '',
0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('name', 'Ic_blood',
0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('equation',
'd(Ic_blood,t)-R_Ic_blood', 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('initialValueU',
'Ic_blood_in', 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('initialValueUt',
0, 0, 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('description', '',
0, 0);
model.component('comp1').physics('ge').feature('ge1').set('DependentVariableQuanti
ty', 'concentration');
```

```
model.component('comp1').physics('ge').feature('ge1').set('CustomSourceTermUnit',
'1');
model.component('comp1').physics('ge').feature('ge1').set('SourceTermQuantity',
'none');
model.component('comp1').physics('ge').feature('ge1').setIndex('CustomSourceTermUn
it', 'mol/s/m^3', 0, 0);
model.component('comp1').physics.create('dode', 'DomainODE', {'u'});

model.study('std1').feature('time').activate('dode', true);

model.component('comp1').physics('dode').field('dimensionless').field('R_lg');
model.component('comp1').physics('dode').field('dimensionless').field('lg');
model.component('comp1').physics('dode').field('dimensionless').component(1,
'lg');
model.component('comp1').physics('dode').tag('lg');
model.component('comp1').physics('lg').feature('dode1').setIndex('f', 'R_lg', 0);
model.component('comp1').physics.create('ale', 'MovingMesh', 'geom1');

model.study('std1').feature('time').activate('ale', true);

model.component('comp1').physics('ale').create('pres1', 'PrescribedDeformation',
1);
model.component('comp1').physics('ale').feature('pres1').setIndex('dx', 'X*lg-X',
0);

model.component('comp1').geom('geom1').create('i1', 'Interval');
model.component('comp1').geom('geom1').feature('i1').setIndex('coord', 0, 0);
model.component('comp1').geom('geom1').feature('i1').setIndex('coord', 0.01, 1);
model.component('comp1').geom('geom1').feature('i1').setIndex('coord', 0, 0);
model.component('comp1').geom('geom1').feature('i1').setIndex('coord', 0.01, 1);
model.component('comp1').geom('geom1').run('i1');
model.component('comp1').geom('geom1').create('i2', 'Interval');
model.component('comp1').geom('geom1').feature('i2').setIndex('coord', 0, 0);
model.component('comp1').geom('geom1').feature('i2').setIndex('coord', 'TumorX',
1);
model.component('comp1').geom('geom1').feature('i2').setIndex('coord', 0, 0);
model.component('comp1').geom('geom1').feature('i2').setIndex('coord', 'TumorX',
1);

model.param.set('kh_T', '4.13e-8[cm^2/(mmHg*s)]');
model.param.descr('kh_T', '');
model.param.set('Pv', '15.6[mmHg]');
model.param.descr('Pv', '');
model.param.set('lplSvH', '0.05 [1/(mmHg*s)]');
model.param.descr('lplSvH', '');
model.param.set('pl', '0');
model.param.descr('pl', '');
model.param.set('kh_H', '8.53e-9[cm^2/(mmHg*s)]');
model.param.descr('kh_H', '');
model.param.set('SvH', '70[1/cm]');
model.param.descr('SvH', '');
model.param.set('SvT', '50[1/cm]');
model.param.descr('SvT', '');
model.param.set('V_blood', '2e-3[L]');
model.param.descr('V_blood', '');
model.param.set('sks21', '4.0180');
model.param.descr('sks21', '');
model.param.set('skt21', '-22.5083');
model.param.descr('skt21', '');
```

```
model.param.set('Lwv', '5e-6[m]');
model.param.descr('Lwv', '');
model.param.set('Vis', '7e-4[Pa*s]');
model.param.descr('Vis', '');
model.param.set('kd', '0.417[d]');
model.param.descr('kd', '');
model.param.set('KbT', '1.38e-23[(m^2*kg)/(s^2*K)]');
model.param.descr('KbT', '');
model.param.set('Tabs', '310.0[K]');
model.param.descr('Tabs', '');
model.param.set('roT', '100[nm]');
model.param.descr('roT', '');
model.param.set('roH', '3.5[nm]');
model.param.descr('roH', '');
model.param.set('lpT', '(gammaT*roT^2)/(8*Vis*Lwv)');
model.param.descr('lpT', '');
model.param.set('lpH', '(gammaH*roH^2)/(8*Vis*Lwv)');
model.param.descr('lpH', '');
model.param.set('gammaH', '1e-4');
model.param.descr('gammaH', '');
model.param.set('gammaT', '1e-3');
model.param.descr('gammaT', '');
model.param.set('DoIc', '(k_B_const*Tabs)/(6*pi*Vis*rsIc)');
model.param.descr('DoIc', '');
model.param.set('rsAb', '1[nm]');
model.param.descr('rsAb', '');
model.param.set('DoAb', '(k_B_const*Tabs)/(6*pi*Vis*rsAb)');
model.param.descr('DoAb', '');
model.param.set('TumorX', 'sqrt(A0/(pi))');
model.param.descr('TumorX', '');
model.param.set('M_w', '121[kg/mol]');
model.param.descr('M_w', '');
model.param.set('Thld', '200');
model.param.descr('Thld', '');
model.param.set('tra', '400');
model.param.descr('tra', '');
model.param.set('rsIc', '(10^(-0.31+0.43*log10(M_w[mol/kg])))*10^(-9)[m]');
model.param.descr('rsIc', '');
model.param.set('A0', '17.33333333[mm^2]');
model.param.descr('A0', '');
model.param.set('kh_T', '4.13e-8[cm^2/(mmHg*s)]');
model.param.descr('kh_T', '');
model.param.set('Pv', '15.6[mmHg]');
model.param.descr('Pv', '');
model.param.set('lplSvH', '0.05 [1/(mmHg*s)]');
model.param.descr('lplSvH', '');
model.param.set('pl', '0');
model.param.descr('pl', '');
model.param.set('kh_H', '8.53e-9[cm^2/(mmHg*s)]');
model.param.descr('kh_H', '');
model.param.set('SvH', '70[1/cm]');
model.param.descr('SvH', '');
model.param.set('SvT', '50[1/cm]');
model.param.descr('SvT', '');
model.param.set('V_blood', '2e-3[L]');
model.param.descr('V_blood', '');
model.param.set('sks21', '4.0180');
model.param.descr('sks21', '');
model.param.set('skt21', '-22.5083');
```

```
model.param.descr('skt21', '');
model.param.set('Lwv', '5e-6[m]');
model.param.descr('Lwv', '');
model.param.set('Vis', '7e-4[Pa*s]');
model.param.descr('Vis', '');
model.param.set('kd', '0.417[d]');
model.param.descr('kd', '');
model.param.set('KbT', '1.38e-23[(m^2*kg)/(s^2*K)]');
model.param.descr('KbT', '');
model.param.set('Tabs', '310.0[K]');
model.param.descr('Tabs', '');
model.param.set('roT', '100[nm]');
model.param.descr('roT', '');
model.param.set('roH', '3.5[nm]');
model.param.descr('roH', '');
model.param.set('lpT', '(gammaT*roT^2)/(8*Vis*Lwv)');
model.param.descr('lpT', '');
model.param.set('lpH', '(gammaH*roH^2)/(8*Vis*Lwv)');
model.param.descr('lpH', '');
model.param.set('gammaH', '1e-4');
model.param.descr('gammaH', '');
model.param.set('gammaT', '1e-3');
model.param.descr('gammaT', '');
model.param.set('DoIc', '(k_B_const*Tabs)/(6*pi*Vis*rsIc)');
model.param.descr('DoIc', '');
model.param.set('rsAb', '1[nm]');
model.param.descr('rsAb', '');
model.param.set('DoAb', '(k_B_const*Tabs)/(6*pi*Vis*rsAb)');
model.param.descr('DoAb', '');
model.param.set('TumorX', 'sqrt(A0/(pi))');
model.param.descr('TumorX', '');
model.param.set('M_w', '121[kg/mol]');
model.param.descr('M_w', '');
model.param.set('Thld', '200');
model.param.descr('Thld', '');
model.param.set('tra', '400');
model.param.descr('tra', '');
model.param.set('rsIc', '(10^(-0.31+0.43*log10(M_w[mol/kg])))*10^(-9)[m]');
model.param.descr('rsIc', '');
model.param.set('A0', '17.33333333[mm^2]');
model.param.descr('A0', '');
model.param.set('k_In', '3e-8[1/day]');
model.param.descr('k_In', '');
model.param.set('k_T_E', '3e-8[1/day]');
model.param.descr('k_T_E', '');
model.param.set('k_APC', '3e-8[1/day]');
model.param.descr('k_APC', '');
model.param.set('k_Th_E', '3e-8[1/day]');
model.param.descr('k_Th_E', '');
model.param.set('K_cAPC', 'K');
model.param.descr('K_cAPC', '');
model.param.set('K_cIn', 'K');
model.param.descr('K_cIn', '');
model.param.set('K', '0.000913463109500608[g/cm^3]');
model.param.descr('K', '');
model.param.set('k_In', '3e-8[1/day]');
model.param.descr('k_In', '');
model.param.set('k_T_E', '3e-8[1/day]');
model.param.descr('k_T_E', '');
```

```
model.param.set('k_APC', '3e-8[1/day]');
model.param.descr('k_APC', '');
model.param.set('k_Th_E', '3e-8[1/day]');
model.param.descr('k_Th_E', '');
model.param.set('K_cAPC', 'K');
model.param.descr('K_cAPC', '');
model.param.set('K_cIn', 'K');
model.param.descr('K_cIn', '');
model.param.set('K', '0.0009134631095006 08[g/cm^3]');
model.param.descr('K', '');
model.param.set('d_c_c', '1.38[1/day]');
model.param.descr('d_c_c', '');
model.param.set('d_IAPC_c', '0.1[1/day]');
model.param.descr('d_IAPC_c', '');
model.param.set('d_APC_c', '0.1[1/day]');
model.param.descr('d_APC_c', '');
model.param.set('d_Th_E_c', '0.18[1/day]');
model.param.descr('d_Th_E_c', '');
model.param.set('d_In_c', '0.18[1/day]');
model.param.descr('d_In_c', '');
model.param.set('d_T_E_c', '0.18[1/day]');
model.param.descr('d_T_E_c', '');
model.param.set('d_Ic_flH', '1.38[1/day]');
model.param.descr('d_Ic_flH', '');
model.param.set('d_clear', '0.4[1/hour]');
model.param.descr('d_clear', '');
model.param.set('d_c_c', '1.38[1/day]');
model.param.descr('d_c_c', '');
model.param.set('d_IAPC_c', '0.1[1/day]');
model.param.descr('d_IAPC_c', '');
model.param.set('d_APC_c', '0.1[1/day]');
model.param.descr('d_APC_c', '');
model.param.set('d_Th_E_c', '0.18[1/day]');
model.param.descr('d_Th_E_c', '');
model.param.set('d_In_c', '0.18[1/day]');
model.param.descr('d_In_c', '');
model.param.set('d_T_E_c', '0.18[1/day]');
model.param.descr('d_T_E_c', '');
model.param.set('d_Ic_flH', '1.38[1/day]');
model.param.descr('d_Ic_flH', '');
model.param.set('d_clear', '0.4[1/hour]');
model.param.descr('d_clear', '');
model.param.set('n_In', 'n');
model.param.descr('n_In', '');
model.param.set('n_ad', 'n*2');
model.param.descr('n_ad', '');
model.param.set('n_Ag', 'n*2');
model.param.descr('n_Ag', '');
model.param.set('n', '54.1856729318554[(cm^3)/(g*day)]');
model.param.descr('n', '');
model.param.set('n_In', 'n');
model.param.descr('n_In', '');
model.param.set('n_ad', 'n*2');
model.param.descr('n_ad', '');
model.param.set('n_Ag', 'n*2');
model.param.descr('n_Ag', '');
model.param.set('n', '54.1856729318554[(cm^3)/(g*day)]');
model.param.descr('n', '');
model.param.set('l_In', 'l');
```

```
model.param.descr('l_In', '');
model.param.set('l_IAPC', 'l');
model.param.descr('l_IAPC', '');
model.param.set('x_APC', '0.5');
model.param.descr('x_APC', '');
model.param.set('m_APC', '2.36009622567891e-06[1/s]');
model.param.descr('m_APC', '');
model.param.set('S_Ic_f', '0');
model.param.descr('S_Ic_f', '');
model.param.set('l', '0.0249989057228036[g/(day*cm^3)]');
model.param.descr('l', '');
model.param.set('l_In', 'l');
model.param.descr('l_In', '');
model.param.set('l_IAPC', 'l');
model.param.descr('l_IAPC', '');
model.param.set('x_APC', '0.5');
model.param.descr('x_APC', '');
model.param.set('m_APC', '2.36009622567891e-06[1/s]');
model.param.descr('m_APC', '');
model.param.set('S_Ic_f', '0');
model.param.descr('S_Ic_f', '');
model.param.set('l', '0.0249989057228036[g/(day*cm^3)]');
model.param.descr('l', '');
model.param.set('c_c_in', '3e-11[g/cm^3]');
model.param.descr('c_c_in', '');
model.param.set('a_in', '3e-11[g/cm^3]');
model.param.descr('a_in', '');
model.param.set('IAPC_in', '5e-5[g/cm^3]');
model.param.descr('IAPC_in', '');
model.param.set('APC_in', '0');
model.param.descr('APC_in', '');
model.param.set('T_E_in', '0');
model.param.descr('T_E_in', '');
model.param.set('S_in', '9e-4[g/cm^3]');
model.param.descr('S_in', '');
model.param.set('In_in', '9e-4[g/cm^3]');
model.param.descr('In_in', '');
model.param.set('T_inT', '0.4[g/cm^3]');
model.param.descr('T_inT', '');
model.param.set('Th_E_in', '0');
model.param.descr('Th_E_in', '');
model.param.set('Ag_In', '0');
model.param.descr('Ag_In', '');
model.param.set('Ab_In', '0');
model.param.descr('Ab_In', '');
model.param.set('Ic_blood_in', '7.3975e-5[mol/m^3]');
model.param.descr('Ic_blood_in', '');
model.param.set('c_c_in', '3e-11[g/cm^3]');
model.param.descr('c_c_in', '');
model.param.set('a_in', '3e-11[g/cm^3]');
model.param.descr('a_in', '');
model.param.set('IAPC_in', '5e-5[g/cm^3]');
model.param.descr('IAPC_in', '');
model.param.set('APC_in', '0');
model.param.descr('APC_in', '');
model.param.set('T_E_in', '0');
model.param.descr('T_E_in', '');
model.param.set('S_in', '9e-4[g/cm^3]');
model.param.descr('S_in', '');
```

```
model.param.set('In_in', '9e-4[g/cm^3]');
model.param.descr('In_in', '');
model.param.set('T_inT', '0.4[g/cm^3]');
model.param.descr('T_inT', '');
model.param.set('Th_E_in', '0');
model.param.descr('Th_E_in', '');
model.param.set('Ag_In', '0');
model.param.descr('Ag_In', '');
model.param.set('Ab_In', '0');
model.param.descr('Ab_In', '');
model.param.set('Ic_blood_in', '7.3975e-5[mol/m^3]');
model.param.descr('Ic_blood_in', '');
model.param.set('Dox', '1.55e-4[m^2/day]');
model.param.descr('Dox', '');
model.param.set('Aox', '2200[mol/m^3/day]');
model.param.descr('Aox', '');
model.param.set('kox', '0.00464[mol/m^3]');
model.param.descr('kox', '');
model.param.set('Perox', 'Dox/5e-6[m]');
model.param.descr('Perox', '');
model.param.set('Ciox', '0.2[mol/m^3]');
model.param.descr('Ciox', '');
model.param.set('k1', '0.344645847342439[1/day]');
model.param.descr('k1', '');
model.param.set('k2', '0.0083[mol/m^3]');
model.param.descr('k2', '');
model.param.set('Dox', '1.55e-4[m^2/day]');
model.param.descr('Dox', '');
model.param.set('Aox', '2200[mol/m^3/day]');
model.param.descr('Aox', '');
model.param.set('kox', '0.00464[mol/m^3]');
model.param.descr('kox', '');
model.param.set('Perox', 'Dox/5e-6[m]');
model.param.descr('Perox', '');
model.param.set('Ciox', '0.2[mol/m^3]');
model.param.descr('Ciox', '');
model.param.set('k1', '0.344645847342439[1/day]');
model.param.descr('k1', '');
model.param.set('k2', '0.0083[mol/m^3]');
model.param.descr('k2', '');
model.param.set('D_c_c', '6.0472e-2[cm^2/day]');
model.param.descr('D_c_c', '');
model.param.set('D_IAPC', '5.11e-13[m^2/s]');
model.param.descr('D_IAPC', '');
model.param.set('D_APC', '5.11e-13[m^2/s]');
model.param.descr('D_APC', '');
model.param.set('D_T_E', '5.11e-13[m^2/s]');
model.param.descr('D_T_E', '');
model.param.set('D_Th_E', '5.11e-13[m^2/s]');
model.param.descr('D_Th_E', '');
model.param.set('D_S', '5.11e-13[m^2/s]');
model.param.descr('D_S', '');
model.param.set('D_In', '5.11e-13[m^2/s]');
model.param.descr('D_In', '');
model.param.set('D_T', '5.11e-13[m^2/s]');
model.param.descr('D_T', '');
model.param.set('D_Ag', '6.0472e-2[cm^2/day]');
model.param.descr('D_Ag', '');
model.param.set('D_Ic_f', '0.5e-6[cm^2/s]');
```

```
model.param.descr('D_Ic_f', '');
model.param.set('D_Ab', '6.0472e-2[cm^2/day]');
model.param.descr('D_Ab', '');
model.param.set('D_c_c', '6.0472e-2[cm^2/day]');
model.param.descr('D_c_c', '');
model.param.set('D_IAPC', '5.11e-13[m^2/s]');
model.param.descr('D_IAPC', '');
model.param.set('D_APC', '5.11e-13[m^2/s]');
model.param.descr('D_APC', '');
model.param.set('D_T_E', '5.11e-13[m^2/s]');
model.param.descr('D_T_E', '');
model.param.set('D_Th_E', '5.11e-13[m^2/s]');
model.param.descr('D_Th_E', '');
model.param.set('D_S', '5.11e-13[m^2/s]');
model.param.descr('D_S', '');
model.param.set('D_In', '5.11e-13[m^2/s]');
model.param.descr('D_In', '');
model.param.set('D_T', '5.11e-13[m^2/s]');
model.param.descr('D_T', '');
model.param.set('D_Ag', '6.0472e-2[cm^2/day]');
model.param.descr('D_Ag', '');
model.param.set('D_Ic_f', '0.5e-6[cm^2/s]');
model.param.descr('D_Ic_f', '');
model.param.set('D_Ab', '6.0472e-2[cm^2/day]');
model.param.descr('D_Ab', '');
model.param.set('k_on', '1e2[m^3/mol/s]');
model.param.descr('k_on', '');
model.param.set('k_off', '1e-3[1/s]');
model.param.descr('k_off', '');
model.param.set('K_T', '1.365e-18 [g/cm^3]');
model.param.descr('K_T', '');
model.param.set('fivf', '0.3');
model.param.descr('fivf', '');
model.param.set('ce', '2e-4[mol/m^3]');
model.param.descr('ce', '');
model.param.set('A_b0', '0');
model.param.descr('A_b0', '');
model.param.set('a', '1');
model.param.descr('a', '');
model.param.set('xZ', '1e2');
model.param.descr('xZ', '');
model.param.set('k_on', '1e2[m^3/mol/s]');
model.param.descr('k_on', '');
model.param.set('k_off', '1e-3[1/s]');
model.param.descr('k_off', '');
model.param.set('K_T', '1.365e-18 [g/cm^3]');
model.param.descr('K_T', '');
model.param.set('fivf', '0.3');
model.param.descr('fivf', '');
model.param.set('ce', '2e-4[mol/m^3]');
model.param.descr('ce', '');
model.param.set('A_b0', '0');
model.param.descr('A_b0', '');
model.param.set('a', '1');
model.param.descr('a', '');
model.param.set('xZ', '1e2');
model.param.descr('xZ', '');
model.param.set('lamdaIcT', 'rsIc/roT');
model.param.descr('lamdaIcT', '');
```

```
model.param.set('WIcT', '(FiIcT*(2-FiIcT)*ksIcT/(2*ktIcT))*(lamdaIcT<1)');
model.param.descr('WIcT', '');
model.param.set('ksIcT1', '(9/4)*(pi^2)*sqrt(2)*((1-lamdaIcT)^(-
5/2))*(1+(7/60)*(1-lamdaIcT)+(2227/50400)*((1-lamdaIcT)^(2)))-4.0180-
3.9788*lamdaIcT-1.9215*(lamdaIcT^2)-4.392*(lamdaIcT^3)+5.006*(lamdaIcT^4)');
model.param.descr('ksIcT1', '');
model.param.set('ktIcT1', '(9/4)*(pi^2)*sqrt(2)*((1-lamdaIcT)^(-5/2))*(1-
(73/60)*(1-lamdaIcT)+(77293/50400)*((1-lamdaIcT)^(2)))-22.5083-5.6117*lamdaIcT-
0.3363*(lamdaIcT^2)-1.216*(lamdaIcT^3)+1.647*(lamdaIcT^4)');
model.param.descr('ktIcT1', '');
model.param.set('sfIcT', '1-WIcT');
model.param.descr('sfIcT', '');
model.param.set('FiIcT', '((1-lamdaIcT)^2)*(lamdaIcT<1)');
model.param.descr('FiIcT', '');
model.param.set('HIcT', '(6*pi*FiIcT/ktIcT)*(lamdaIcT<1)');
model.param.descr('HIcT', '');
model.param.set('PerIcT', '((gammaT*HIcT*DoIc)/Lwv)*(lamdaIcT<1)');
model.param.descr('PerIcT', '');
model.param.set('ksIcT', 'if(lamdaIcT<1,ksIcT1,1)');
model.param.descr('ksIcT', '');
model.param.set('ktIcT', 'if(lamdaIcT<1,ktIcT1,1)');
model.param.descr('ktIcT', '');
model.param.set('lamdaIcT', 'rsIc/roT');
model.param.descr('lamdaIcT', '');
model.param.set('WIcT', '(FiIcT*(2-FiIcT)*ksIcT/(2*ktIcT))*(lamdaIcT<1)');
model.param.descr('WIcT', '');
model.param.set('ksIcT1', '(9/4)*(pi^2)*sqrt(2)*((1-lamdaIcT)^(-
5/2))*(1+(7/60)*(1-lamdaIcT)+(2227/50400)*((1-lamdaIcT)^(2)))-4.0180-
3.9788*lamdaIcT-1.9215*(lamdaIcT^2)-4.392*(lamdaIcT^3)+5.006*(lamdaIcT^4)');
model.param.descr('ksIcT1', '');
model.param.set('ktIcT1', '(9/4)*(pi^2)*sqrt(2)*((1-lamdaIcT)^(-5/2))*(1-
(73/60)*(1-lamdaIcT)+(77293/50400)*((1-lamdaIcT)^(2)))-22.5083-5.6117*lamdaIcT-
0.3363*(lamdaIcT^2)-1.216*(lamdaIcT^3)+1.647*(lamdaIcT^4)');
model.param.descr('ktIcT1', '');
model.param.set('sfIcT', '1-WIcT');
model.param.descr('sfIcT', '');
model.param.set('FiIcT', '((1-lamdaIcT)^2)*(lamdaIcT<1)');
model.param.descr('FiIcT', '');
model.param.set('HIcT', '(6*pi*FiIcT/ktIcT)*(lamdaIcT<1)');
model.param.descr('HIcT', '');
model.param.set('PerIcT', '((gammaT*HIcT*DoIc)/Lwv)*(lamdaIcT<1)');
model.param.descr('PerIcT', '');
model.param.set('ksIcT', 'if(lamdaIcT<1,ksIcT1,1)');
model.param.descr('ksIcT', '');
model.param.set('ktIcT', 'if(lamdaIcT<1,ktIcT1,1)');
model.param.descr('ktIcT', '');
model.param.set('lamdaIcH', 'rsIc/roH');
model.param.descr('lamdaIcH', '');
model.param.set('WIcH', '(FiIcH*(2-FiIcH)*ksIcH/(2*ktIcH))*(lamdaIcH<1)');
model.param.descr('WIcH', '');
model.param.set('sfIcH', '1-WIcH');
model.param.descr('sfIcH', '');
model.param.set('FiIcH', '((1-lamdaIcH)^2)*(lamdaIcH<1)');
model.param.descr('FiIcH', '');
model.param.set('HIcH', '(6*pi*FiIcH/ktIcH)*(lamdaIcH<1)');
model.param.descr('HIcH', '');
model.param.set('PerIcH', '((gammaH*HIcH*DoIc)/Lwv)*(lamdaIcH<1)');
model.param.descr('PerIcH', '');
```

```
model.param.set('ksIcH1', '((9/4)*(pi^2)*sqrt(2)*((1-lamdaIcH)^(-
5/2))*(1+(7/60)*(1-lamdaIcH)+(2227/50400)*((1-lamdaIcH)^(2)))-4.0180-
3.9788*lamdaIcH-1.9215*(lamdaIcH^2)-4.392*(lamdaIcH^3)+5.006*(lamdaIcH^4))');
model.param.descr('ksIcH1', '');
model.param.set('ktIcH1', '((9/4)*(pi^2)*sqrt(2)*((1-lamdaIcH)^(-5/2))*(1-
(73/60)*(1-lamdaIcH)+(77293/50400)*((1-lamdaIcH)^(2)))-22.5083-5.6117*lamdaIcH-
0.3363*(lamdaIcH^2)-1.216*(lamdaIcH^3)+1.647*(lamdaIcH^4))');
model.param.descr('ktIcH1', '');
model.param.set('ksIcH', 'if(lamdaIcH<1,ksIcH1,1)');
model.param.descr('ksIcH', '');
model.param.set('ktIcH', 'if(lamdaIcH<1,ktIcH1,1)');
model.param.descr('ktIcH', '');
model.param.set('lamdaIcH', 'rsIc/roH');
model.param.descr('lamdaIcH', '');
model.param.set('WIcH', '(FiIcH*(2-FiIcH)*ksIcH/(2*ktIcH))*(lamdaIcH<1)');
model.param.descr('WIcH', '');
model.param.set('sfIcH', '1-WIcH');
model.param.descr('sfIcH', '');
model.param.set('FiIcH', '((1-lamdaIcH)^2)*(lamdaIcH<1)');
model.param.descr('FiIcH', '');
model.param.set('HIcH', '(6*pi*FiIcH/ktIcH)*(lamdaIcH<1)');
model.param.descr('HIcH', '');
model.param.set('PerIcH', '((gammaH*HIcH*DoIc)/Lwv)*(lamdaIcH<1)');
model.param.descr('PerIcH', '');
model.param.set('ksIcH1', '((9/4)*(pi^2)*sqrt(2)*((1-lamdaIcH)^(-
5/2))*(1+(7/60)*(1-lamdaIcH)+(2227/50400)*((1-lamdaIcH)^(2)))-4.0180-
3.9788*lamdaIcH-1.9215*(lamdaIcH^2)-4.392*(lamdaIcH^3)+5.006*(lamdaIcH^4))');
model.param.descr('ksIcH1', '');
model.param.set('ktIcH1', '((9/4)*(pi^2)*sqrt(2)*((1-lamdaIcH)^(-5/2))*(1-
(73/60)*(1-lamdaIcH)+(77293/50400)*((1-lamdaIcH)^(2)))-22.5083-5.6117*lamdaIcH-
0.3363*(lamdaIcH^2)-1.216*(lamdaIcH^3)+1.647*(lamdaIcH^4))');
model.param.descr('ktIcH1', '');
model.param.set('ksIcH', 'if(lamdaIcH<1,ksIcH1,1)');
model.param.descr('ksIcH', '');
model.param.set('ktIcH', 'if(lamdaIcH<1,ktIcH1,1)');
model.param.descr('ktIcH', '');

model.component('comp1').geom('geom1').run('fin');

model.component('comp1').variable.create('var1');
model.component('comp1').variable('var1').set('uf', '-kh*d(p,x)+xt');
model.component('comp1').variable('var1').descr('uf', '');
model.component('comp1').variable('var1').set('c', 'c_c+(Ic_f+Ic_b)*M_w');
model.component('comp1').variable('var1').descr('c', '');
model.component('comp1').variable('var1').set('Tregion', 'T[m^3/kg]>200');
model.component('comp1').variable('var1').descr('Tregion', '');
model.component('comp1').variable('var1').set('TV2', '4*pi*inte(Tregion*x^2)');
model.component('comp1').variable('var1').descr('TV2', '');
model.component('comp1').variable('var1').set('TR', 'inte(Tregion)');
model.component('comp1').variable('var1').descr('TR', '');
model.component('comp1').variable('var1').set('TA', 'pi*TR^2');
model.component('comp1').variable('var1').descr('TA', '');
model.component('comp1').variable('var1').set('TV', '(4/3)*pi*TR^3');
model.component('comp1').variable('var1').descr('TV', '');
model.component('comp1').variable('var1').set('ExpTrea', 'int1(t)');
model.component('comp1').variable('var1').descr('ExpTrea', '');
model.component('comp1').variable('var1').set('ExpUntr', 'int2(t)');
model.component('comp1').variable('var1').descr('ExpUntr', '');
model.component('comp1').variable('var1').set('ExpTrea2', 'int4(t)');
```

```
model.component('comp1').variable('var1').descr('ExpTrea2', '');
model.component('comp1').variable('var1').set('ExpUntr2', 'int3(t)');
model.component('comp1').variable('var1').descr('ExpUntr2', '');
model.component('comp1').variable('var1').set('Q_Ic', '(PerIc*Sv*(Ic_blood-
Ic_f))+lp*Sv*(Pv-p)*(1-sfIc)*if(Pv>p, Ic_blood,Ic_f)-d_Ic_fl*Ic_f');
model.component('comp1').variable('var1').descr('Q_Ic', '');
model.component('comp1').variable('var1').set('Pe', 'if(PerIc==0,0,(lp*(Pv-p)*(1-
sfIc))/PerIc)');
model.component('comp1').variable('var1').descr('Pe', '');
model.component('comp1').variable('var1').set('Ic_f_in', 'int6(X)');
model.component('comp1').variable('var1').descr('Ic_f_in', '');
model.component('comp1').variable('var1').set('Ic_b_in', 'int5(X)');
model.component('comp1').variable('var1').descr('Ic_b_in', '');
model.component('comp1').variable('var1').set('Svf', '0.02734*Sv*1[cm]-0.9138');
model.component('comp1').variable('var1').descr('Svf', '');
model.component('comp1').variable('var1').set('uf', '-kh*d(p,x)+xt');
model.component('comp1').variable('var1').descr('uf', '');
model.component('comp1').variable('var1').set('c', 'c_c+(Ic_f+Ic_b)*M_w');
model.component('comp1').variable('var1').descr('c', '');
model.component('comp1').variable('var1').set('Tregion', 'T[m^3/kg]>200');
model.component('comp1').variable('var1').descr('Tregion', '');
model.component('comp1').variable('var1').set('TV2', '4*pi*inte(Tregion*x^2)');
model.component('comp1').variable('var1').descr('TV2', '');
model.component('comp1').variable('var1').set('TR', 'inte(Tregion)');
model.component('comp1').variable('var1').descr('TR', '');
model.component('comp1').variable('var1').set('TA', 'pi*TR^2');
model.component('comp1').variable('var1').descr('TA', '');
model.component('comp1').variable('var1').set('TV', '(4/3)*pi*TR^3');
model.component('comp1').variable('var1').descr('TV', '');
model.component('comp1').variable('var1').set('ExpTrea', 'int1(t)');
model.component('comp1').variable('var1').descr('ExpTrea', '');
model.component('comp1').variable('var1').set('ExpUntr', 'int2(t)');
model.component('comp1').variable('var1').descr('ExpUntr', '');
model.component('comp1').variable('var1').set('ExpTrea2', 'int4(t)');
model.component('comp1').variable('var1').descr('ExpTrea2', '');
model.component('comp1').variable('var1').set('ExpUntr2', 'int3(t)');
model.component('comp1').variable('var1').descr('ExpUntr2', '');
model.component('comp1').variable('var1').set('Q_Ic', '(PerIc*Sv*(Ic_blood-
Ic_f))+lp*Sv*(Pv-p)*(1-sfIc)*if(Pv>p, Ic_blood,Ic_f)-d_Ic_fl*Ic_f');
model.component('comp1').variable('var1').descr('Q_Ic', '');
model.component('comp1').variable('var1').set('Pe', 'if(PerIc==0,0,(lp*(Pv-p)*(1-
sfIc))/PerIc)');
model.component('comp1').variable('var1').descr('Pe', '');
model.component('comp1').variable('var1').set('Ic_f_in', 'int6(X)');
model.component('comp1').variable('var1').descr('Ic_f_in', '');
model.component('comp1').variable('var1').set('Ic_b_in', 'int5(X)');
model.component('comp1').variable('var1').descr('Ic_b_in', '');
model.component('comp1').variable('var1').set('Svf', '0.02734*Sv*1[cm]-0.9138');
model.component('comp1').variable('var1').descr('Svf', '');
model.component('comp1').variable('var1').set('R_In', 'Svf*(l_In*c/(K_cIn+c))-
d_In*In');
model.component('comp1').variable('var1').descr('R_In', '');
model.component('comp1').variable('var1').set('R_Ic_f', '-
(k_on*ce*Ic_f/fivf)+k_off*Ic_b+Q_Ic');
model.component('comp1').variable('var1').descr('R_Ic_f', '');
model.component('comp1').variable('var1').set('R_Ic_b', '(k_on*ce*Ic_f/fivf)-
k_off*Ic_b');
model.component('comp1').variable('var1').descr('R_Ic_b', '');
```

```
model.component('comp1').variable('var1').set('R_Ic_blood', '-
4*pi*inte(Q_Ic*x^2)/V_blood-d_clear*Ic_blood');
model.component('comp1').variable('var1').descr('R_Ic_blood', '');
model.component('comp1').variable('var1').set('R_T', '((k1*cox)/(cox+k2))*T-
(n_In*In+n_In*IAPC+n_ad*T_E)*T');
model.component('comp1').variable('var1').descr('R_T', '');
model.component('comp1').variable('var1').set('R_Ag', '(n_In*In+n_ad*T_E)*T-
n_Ag*IAPC*Ag');
model.component('comp1').variable('var1').descr('R_Ag', '');
model.component('comp1').variable('var1').set('R_Th_E', 'Svf*m_APC*APC-
d_Th_E*Th_E');
model.component('comp1').variable('var1').descr('R_Th_E', '');
model.component('comp1').variable('var1').set('R_T_E', 'Svf*m_APC*APC-d_T_E*T_E');
model.component('comp1').variable('var1').descr('R_T_E', '');
model.component('comp1').variable('var1').set('R_APC',
'x_APC*(c/(K_cAPC+c))*(n_In*IAPC*T+n_Ag*IAPC*Ag)-d_APC*APC');
model.component('comp1').variable('var1').descr('R_APC', '');
model.component('comp1').variable('var1').set('R_c_c',
'k_In*In+k_T_E*T_E+k_Th_E*Th_E+k_APC*APC-d_c_c *c_c');
model.component('comp1').variable('var1').descr('R_c_c', '');
model.component('comp1').variable('var1').set('R_IAPC', 'Svf*(l_IAPC*c/(K_cIn+c))-
d_IAPC*IAPC-x_APC*(c/(K_cAPC+c))*(n_In*IAPC*T+n_Ag*IAPC*Ag)');
model.component('comp1').variable('var1').descr('R_IAPC', '');
model.component('comp1').variable('var1').set('R_lg', '(1/3)*lg*R_T/T_inT');
model.component('comp1').variable('var1').descr('R_lg', '');
model.component('comp1').variable('var1').set('R_In', 'Svf*(l_In*c/(K_cIn+c))-
d_In*In');
model.component('comp1').variable('var1').descr('R_In', '');
model.component('comp1').variable('var1').set('R_Ic_f', '-
(k_on*ce*Ic_f/fivf)+k_off*Ic_b+Q_Ic');
model.component('comp1').variable('var1').descr('R_Ic_f', '');
model.component('comp1').variable('var1').set('R_Ic_b', '(k_on*ce*Ic_f/fivf)-
k_off*Ic_b');
model.component('comp1').variable('var1').descr('R_Ic_b', '');
model.component('comp1').variable('var1').set('R_Ic_blood', '-
4*pi*inte(Q_Ic*x^2)/V_blood-d_clear*Ic_blood');
model.component('comp1').variable('var1').descr('R_Ic_blood', '');
model.component('comp1').variable('var1').set('R_T', '((k1*cox)/(cox+k2))*T-
(n_In*In+n_In*IAPC+n_ad*T_E)*T');
model.component('comp1').variable('var1').descr('R_T', '');
model.component('comp1').variable('var1').set('R_Ag', '(n_In*In+n_ad*T_E)*T-
n_Ag*IAPC*Ag');
model.component('comp1').variable('var1').descr('R_Ag', '');
model.component('comp1').variable('var1').set('R_Th_E', 'Svf*m_APC*APC-
d_Th_E*Th_E');
model.component('comp1').variable('var1').descr('R_Th_E', '');
model.component('comp1').variable('var1').set('R_T_E', 'Svf*m_APC*APC-d_T_E*T_E');
model.component('comp1').variable('var1').descr('R_T_E', '');
model.component('comp1').variable('var1').set('R_APC',
'x_APC*(c/(K_cAPC+c))*(n_In*IAPC*T+n_Ag*IAPC*Ag)-d_APC*APC');
model.component('comp1').variable('var1').descr('R_APC', '');
model.component('comp1').variable('var1').set('R_c_c',
'k_In*In+k_T_E*T_E+k_Th_E*Th_E+k_APC*APC-d_c_c *c_c');
model.component('comp1').variable('var1').descr('R_c_c', '');
model.component('comp1').variable('var1').set('R_IAPC', 'Svf*(l_IAPC*c/(K_cIn+c))-
d_IAPC*IAPC-x_APC*(c/(K_cAPC+c))*(n_In*IAPC*T+n_Ag*IAPC*Ag)');
model.component('comp1').variable('var1').descr('R_IAPC', '');
model.component('comp1').variable('var1').set('R_lg', '(1/3)*lg*R_T/T_inT');
model.component('comp1').variable('var1').descr('R_lg', '');
```

```
model.component('comp1').variable('var1').set('Sv', 'stepSv(T[m^3/kg])[1/m]');
model.component('comp1').variable('var1').descr('Sv', '');
model.component('comp1').variable('var1').set('lplSv',
'steplplSv(T[m^3/kg])[m*s/kg]');
model.component('comp1').variable('var1').descr('lplSv', '');
model.component('comp1').variable('var1').set('lp',
'steplp(T[m^3/kg])[m^2*s/kg]');
model.component('comp1').variable('var1').descr('lp', '');
model.component('comp1').variable('var1').set('PerIc',
'stepPerIc(T[m^3/kg])[m/s]');
model.component('comp1').variable('var1').descr('PerIc', '');
model.component('comp1').variable('var1').set('d_Ic_fl',
'stepd_Ic_fl(T[m^3/kg])[1/s]');
model.component('comp1').variable('var1').descr('d_Ic_fl', '');
model.component('comp1').variable('var1').set('sfIc', 'stepsfIc(T[m^3/kg])');
model.component('comp1').variable('var1').descr('sfIc', '');
model.component('comp1').variable('var1').set('kh',
'stepkh(T[m^3/kg])[m^3*s/kg]');
model.component('comp1').variable('var1').descr('kh', '');
model.component('comp1').variable('var1').set('Sv', 'stepSv(T[m^3/kg])[1/m]');
model.component('comp1').variable('var1').descr('Sv', '');
model.component('comp1').variable('var1').set('lplSv',
'steplplSv(T[m^3/kg])[m*s/kg]');
model.component('comp1').variable('var1').descr('lplSv', '');
model.component('comp1').variable('var1').set('lp',
'steplp(T[m^3/kg])[m^2*s/kg]');
model.component('comp1').variable('var1').descr('lp', '');
model.component('comp1').variable('var1').set('PerIc',
'stepPerIc(T[m^3/kg])[m/s]');
model.component('comp1').variable('var1').descr('PerIc', '');
model.component('comp1').variable('var1').set('d_Ic_fl',
'stepd_Ic_fl(T[m^3/kg])[1/s]');
model.component('comp1').variable('var1').descr('d_Ic_fl', '');
model.component('comp1').variable('var1').set('sfIc', 'stepsfIc(T[m^3/kg])');
model.component('comp1').variable('var1').descr('sfIc', '');
model.component('comp1').variable('var1').set('kh',
'stepkh(T[m^3/kg])[m^3*s/kg]');
model.component('comp1').variable('var1').descr('kh', '');
model.component('comp1').variable('var1').set('d_IAPC', 'd_IAPC_c*dox');
model.component('comp1').variable('var1').descr('d_IAPC', '');
model.component('comp1').variable('var1').set('d_APC', 'd_APC_c*dox');
model.component('comp1').variable('var1').descr('d_APC', '');
model.component('comp1').variable('var1').set('d_Th_E', 'd_Th_E_c*dox');
model.component('comp1').variable('var1').descr('d_Th_E', '');
model.component('comp1').variable('var1').set('d_In', 'd_In_c*dox');
model.component('comp1').variable('var1').descr('d_In', '');
model.component('comp1').variable('var1').set('d_T_E', 'd_T_E_c *dox');
model.component('comp1').variable('var1').descr('d_T_E', '');
model.component('comp1').variable('var1').set('dox', '1+1.025*(1-(cox/Ciox))');
model.component('comp1').variable('var1').descr('dox', '');
model.component('comp1').variable('var1').set('d_IAPC', 'd_IAPC_c*dox');
model.component('comp1').variable('var1').descr('d_IAPC', '');
model.component('comp1').variable('var1').set('d_APC', 'd_APC_c*dox');
model.component('comp1').variable('var1').descr('d_APC', '');
model.component('comp1').variable('var1').set('d_Th_E', 'd_Th_E_c*dox');
model.component('comp1').variable('var1').descr('d_Th_E', '');
model.component('comp1').variable('var1').set('d_In', 'd_In_c*dox');
model.component('comp1').variable('var1').descr('d_In', '');
model.component('comp1').variable('var1').set('d_T_E', 'd_T_E_c *dox');
```

```
model.component('comp1').variable('var1').descr('d_T_E', '');
model.component('comp1').variable('var1').set('dox', '1+1.025*(1-(cox/Ciox))');
model.component('comp1').variable('var1').descr('dox', '');
model.component('comp1').variable('var1').set('c_c_T',
'4*pi*inte(c_c*Tregion*x^2)');
model.component('comp1').variable('var1').descr('c_c_T', '');
model.component('comp1').variable('var1').set('IAPC_T',
'4*pi*inte(IAPC*Tregion*x^2)');
model.component('comp1').variable('var1').descr('IAPC_T', '');
model.component('comp1').variable('var1').set('In_T',
'4*pi*inte(In*Tregion*x^2)');
model.component('comp1').variable('var1').descr('In_T', '');
model.component('comp1').variable('var1').set('APC_T',
'4*pi*inte(APC*Tregion*x^2)');
model.component('comp1').variable('var1').descr('APC_T', '');
model.component('comp1').variable('var1').set('Th_E_T',
'4*pi*inte(Th_E*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Th_E_T', '');
model.component('comp1').variable('var1').set('T_E_T',
'4*pi*inte(T_E*Tregion*x^2)');
model.component('comp1').variable('var1').descr('T_E_T', '');
model.component('comp1').variable('var1').set('Ag_T',
'4*pi*inte(Ag*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Ag_T', '');
model.component('comp1').variable('var1').set('T_T', '4*pi*inte(T*Tregion*x^2)');
model.component('comp1').variable('var1').descr('T_T', '');
model.component('comp1').variable('var1').set('cox_T',
'4*pi*inte(cox*Tregion*x^2)');
model.component('comp1').variable('var1').descr('cox_T', '');
model.component('comp1').variable('var1').set('Ic_b_T',
'4*pi*inte(Ic_b*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Ic_b_T', '');
model.component('comp1').variable('var1').set('c_T', '4*pi*inte(c*Tregion*x^2)');
model.component('comp1').variable('var1').descr('c_T', '');
model.component('comp1').variable('var1').set('c_c_T',
'4*pi*inte(c_c*Tregion*x^2)');
model.component('comp1').variable('var1').descr('c_c_T', '');
model.component('comp1').variable('var1').set('IAPC_T',
'4*pi*inte(IAPC*Tregion*x^2)');
model.component('comp1').variable('var1').descr('IAPC_T', '');
model.component('comp1').variable('var1').set('In_T',
'4*pi*inte(In*Tregion*x^2)');
model.component('comp1').variable('var1').descr('In_T', '');
model.component('comp1').variable('var1').set('APC_T',
'4*pi*inte(APC*Tregion*x^2)');
model.component('comp1').variable('var1').descr('APC_T', '');
model.component('comp1').variable('var1').set('Th_E_T',
'4*pi*inte(Th_E*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Th_E_T', '');
model.component('comp1').variable('var1').set('T_E_T',
'4*pi*inte(T_E*Tregion*x^2)');
model.component('comp1').variable('var1').descr('T_E_T', '');
model.component('comp1').variable('var1').set('Ag_T',
'4*pi*inte(Ag*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Ag_T', '');
model.component('comp1').variable('var1').set('T_T', '4*pi*inte(T*Tregion*x^2)');
model.component('comp1').variable('var1').descr('T_T', '');
model.component('comp1').variable('var1').set('cox_T',
'4*pi*inte(cox*Tregion*x^2)');
```

```
model.component('comp1').variable('var1').descr('cox_T', '');
model.component('comp1').variable('var1').set('Ic_b_T',
'4*pi*inte(Ic_b*Tregion*x^2)');
model.component('comp1').variable('var1').descr('Ic_b_T', '');
model.component('comp1').variable('var1').set('c_T', '4*pi*inte(c*Tregion*x^2)');
model.component('comp1').variable('var1').descr('c_T', '');

model.component('comp1').func.create('step1', 'Step');
model.component('comp1').func.create('step2', 'Step');
model.component('comp1').func.create('step3', 'Step');
model.component('comp1').func.create('step4', 'Step');
model.component('comp1').func.create('step5', 'Step');
model.component('comp1').func.create('step6', 'Step');
model.component('comp1').func.create('step7', 'Step');
model.component('comp1').func('step1').set('location', 'TumorX');
model.component('comp1').func('step1').set('from', 'T_inT');
model.component('comp1').func('step1').set('to', 0);
model.component('comp1').func('step1').set('smooth', 'TumorX*2');
model.component('comp1').func('step1').label('StepT_in');
model.component('comp1').func('step1').set('funcname', 'stepT_in');
model.component('comp1').func('step2').label('StepSv');
model.component('comp1').func('step2').set('funcname', 'stepSv');
model.component('comp1').func('step2').set('location', 'Thld');
model.component('comp1').func('step2').set('from', 'SvH');
model.component('comp1').func('step2').set('to', 'SvT');
model.component('comp1').func('step2').set('smooth', 'tra');
model.component('comp1').func('step3').label('SteplplSv');
model.component('comp1').func('step3').set('funcname', 'steplplSv');
model.component('comp1').func('step3').set('location', 'Thld');
model.component('comp1').func('step3').set('from', 'lplSvH');
model.component('comp1').func('step3').set('to', 0);
model.component('comp1').func('step3').set('smooth', 'tra');
model.component('comp1').func('step4').label('Steplp');
model.component('comp1').func('step4').set('funcname', 'steplp');
model.component('comp1').func('step4').set('location', 'Thld');
model.component('comp1').func('step4').set('from', 'lpH');
model.component('comp1').func('step4').set('to', 'lpT');
model.component('comp1').func('step4').set('smooth', 'tra');
model.component('comp1').func('step5').label('StepPerIc');
model.component('comp1').func('step5').set('funcname', 'stepPerIc');
model.component('comp1').func('step5').set('location', 'Thld');
model.component('comp1').func('step5').set('from', 'PerIcH');
model.component('comp1').func('step5').set('to', 'PerIcT');
model.component('comp1').func('step5').set('smooth', 'tra');
model.component('comp1').func('step6').label('Stepd_Ic_fl');
model.component('comp1').func('step6').set('funcname', 'Stepd_Ic_fl');
model.component('comp1').func('step6').set('location', 'Thld');
model.component('comp1').func('step6').set('from', 'd_Ic_flH');
model.component('comp1').func('step6').set('to', 0);
model.component('comp1').func('step6').set('smooth', 'tra');
model.component('comp1').func('step7').label('StepsfIc');
model.component('comp1').func('step7').set('funcname', 'stepsfIc');
model.component('comp1').func('step7').set('location', 'Thld');
model.component('comp1').func('step7').set('from', 'sfIcH');
model.component('comp1').func('step7').set('to', 'sfIcT');
model.component('comp1').func('step7').set('smooth', 'tra');
model.component('comp1').func.create('step8', 'Step');
model.component('comp1').func('step8').label('Stepkh');
model.component('comp1').func('step8').set('funcname', 'stepkh');
```

```
model.component('comp1').func('step8').set('location', 'Thld');
model.component('comp1').func('step8').set('from', 'kh_H');
model.component('comp1').func('step8').set('to', 'kh_T');
model.component('comp1').func('step8').set('smooth', 'tra');

model.component('comp1').selection.create('sel1', 'Explicit');
model.component('comp1').selection('sel1').label('Tumor');
model.component('comp1').selection('sel1').label('Tumor center');
model.component('comp1').selection('sel1').geom(0);
model.component('comp1').selection('sel1').set([1]);
model.component('comp1').selection.create('sel2', 'Explicit');
model.component('comp1').selection('sel2').label('External point');
model.component('comp1').selection('sel2').geom(0);
model.component('comp1').selection('sel2').set([3]);

model.component('comp1').geom('geom1').feature('i2').active(false);
model.component('comp1').geom('geom1').runPre('fin');
model.component('comp1').geom('geom1').run;
model.component('comp1').geom('geom1').run('fin');

model.component('comp1').physics('c_c').feature('gfeq1').setIndex('Ga', '(-
D_c_c*c_cx*x^2)+(uf*c_c*x^2)', 0);
model.component('comp1').physics('c_c').feature('gfeq1').setIndex('f',
'R_c_c*x^2', 0);
model.component('comp1').physics('c_c').feature('gfeq1').setIndex('da', 'x^2', 0);
model.component('comp1').physics('Ic_f').create('init2', 'init', 1);
model.component('comp1').physics('Ic_f').feature('init2').set('Ic_f', 'Ic_f_in');
model.component('comp1').physics('Ic_f').feature('init2').selection.all;
model.component('comp1').physics('Ic_f').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('Ic_f').feature('dir1').selection.named('sel2');
model.component('comp1').physics('Ic_b').feature('init1').set('Ic_b', 'Ic_b_in');
model.component('comp1').physics('Ic_b').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('Ic_b').feature('dir1').selection.named('sel2');
model.component('comp1').physics('c_c').feature('init1').set('c_c', 'c_c_in');
model.component('comp1').physics('IAPC').feature('init1').set('IAPC', 'IAPC_in');
model.component('comp1').physics('APC').feature('init1').set('APC', 'APC_in');
model.component('comp1').physics('APC').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('APC').feature('dir1').selection.named('sel2');
model.component('comp1').physics('g2').tag('Ag');
model.component('comp1').physics('g').tag('p');
model.component('comp1').physics.create('g', 'GeneralFormPDE', {'u'});

model.study('std1').feature('time').activate('g', true);

model.component('comp1').physics('g').prop('Units').set('DependentVariableQuantity
', 'massconcentration');
model.component('comp1').physics('g').feature('gfeq1').setIndex('Ga', '-
D_In*Inx*x^2+xt*In*x^2', 0);
model.component('comp1').physics('g').feature('gfeq1').setIndex('f', 'R_In*x^2',
0);
model.component('comp1').physics('g').feature('init1').set('u', 'In_in');
model.component('comp1').physics('g').field('dimensionless').field('In');
model.component('comp1').physics('g').field('dimensionless').component(1, 'In');
model.component('comp1').physics('g').tag('In');
model.component('comp1').physics('In').prop('Units').setIndex('CustomSourceTermUni
t', 'kg/m/s', 0, 0);
model.component('comp1').physics('ale').feature('pres1').selection.all;
model.component('comp1').physics('Th_E').feature('init1').set('Th_E', 'Th_E_in');
model.component('comp1').physics('Th_E').create('dir1', 'DirichletBoundary', 0);
```

```
model.component('comp1').physics('Th_E').feature('dir1').selection.named('sel2');
model.component('comp1').physics('T_E').feature('init1').set('T_E', 'T_E_in');
model.component('comp1').physics('T_E').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('T_E').feature('dir1').selection.named('sel2');
model.component('comp1').physics('Ag').feature('init1').set('Ag', 'Ag_In');
model.component('comp1').physics('Ag').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('Ag').feature('dir1').selection.named('sel2');
model.component('comp1').physics('T').feature('init1').set('T',
'stepT_in(X[1/m])');
model.component('comp1').physics('T').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('T').feature('dir1').selection.named('sel2');
model.component('comp1').physics('p').feature('init1').set('p', '15.6[mmHg]');
model.component('comp1').physics('cox').feature('init1').set('cox', 0.2);
model.component('comp1').physics('cox').prop('Units').set('DependentVariableQuanti
ty', 'concentration');
model.component('comp1').physics('cox').prop('Units').setIndex('CustomSourceTermUn
it', 'mol/m/s', 0, 0);
model.component('comp1').physics('cox').create('dir1', 'DirichletBoundary', 0);
model.component('comp1').physics('cox').feature('dir1').setIndex('r', 'Ciox', 0);
model.component('comp1').physics('cox').feature('dir1').selection.named('sel2');

model.component('comp1').mesh('mesh1').create('size1', 'Size');
model.component('comp1').mesh('mesh1').feature('size').set('hauto', 1);
model.component('comp1').mesh('mesh1').feature.remove('size1');
model.component('comp1').mesh('mesh1').create('dis1', 'Distribution');
model.component('comp1').mesh('mesh1').feature('dis1').selection.all;
model.component('comp1').mesh('mesh1').feature('dis1').set('type', 'predefined');
model.component('comp1').mesh('mesh1').feature('dis1').set('elemcount', 100);
model.component('comp1').mesh('mesh1').feature('dis1').set('reverse', true);
model.component('comp1').mesh('mesh1').feature('dis1').set('method',
'arithmetic');
model.component('comp1').mesh('mesh1').run;
model.component('comp1').mesh('mesh1').create('edg1', 'Edge');
model.component('comp1').mesh('mesh1').feature('edg1').selection.geom('geom1', 1);
model.component('comp1').mesh('mesh1').feature('edg1').selection.all;
model.component('comp1').mesh('mesh1').run;

model.study('std1').feature('time').set('tlist', 'range(0,0.005,10)');
model.study('std1').feature('time').set('tunit', 'd');
model.study('std1').feature('time').set('usertol', true);
model.study('std1').feature('time').set('rtol', '0.0001');

model.component('comp1').physics('lg').prop('Units').setIndex('CustomSourceTermUni
t', '1/s', 0, 0);
model.component('comp1').physics('lg').feature('init1').set('lg', 1);

model.component('comp1').cpl.create('intop1', 'Integration');
model.component('comp1').cpl('intop1').set('axisym', true);
model.component('comp1').cpl('intop1').set('opname', 'inte');
model.component('comp1').cpl('intop1').selection.all;

model.component('comp1').func('step6').set('funcname', 'stepd_Ic_fl');

model.component('comp1').physics('T').feature('gfeq1').setIndex('da', 1, 0);
model.component('comp1').physics('T_E').feature('gfeq1').setIndex('da', 1, 0);
model.component('comp1').physics('Ic_b').feature('init1').set('Ic_b', 0);
model.component('comp1').physics('Ic_f').feature('init2').set('Ic_f', 0);
model.component('comp1').physics('ge').feature('ge1').setIndex('initialValueU', 0,
0, 0);
```

```
out = model;
```