Parameter Estimation for a Chemical Reaction

Introduction

This application estimates the rate parameters for a reversible reaction with dimerization of an intermediate.

$$\mathrm{A} + \mathrm{B} \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{C}$$

$$\mathrm{C} + \mathrm{C} \underset{k_4}{\overset{k_3}{\rightleftharpoons}} \mathrm{D}$$

It does this by

- parameterizing (with respect to k_1, k_2, k_3 and k_4) the numerical solution of the different equations that describe the reaction kinetics
- calculating the sum of the square of the errors between the model predictions and experimental data
- minimizing the sum of the square of the errors to find the best fit values of k_1, k_2, k_3 and k_4.

```
> restart : with( plots ) : with( Optimization ) :
```

Parameters and Experimental Data

```
> A_0 := 2.1 :
  B_0 := 3.1 :
```

Concentrations of C and D over time

```
> times := [ 0  7  14  21  28  35  42  49  56  63  70 ] :
> C_exp := [ 0  1.065  1.383  0.9793  1.107  0.7289  0.7236  0.4674  0.6031  0.6149  0.3369 ] :
> D_exp := [ 0  0.0058  0.2203  0.4019  0.3638  0.456  0.5014  0.715  0.4723  0.7219  0.7294 ] :
```

Reaction Kinetics

```
> de1 := d/dt C_C(t) = k_1·( A_0 − C_C(t) − 2 C_D(t) )·( B_0 − C_C(t) − 2·C_D(t) ) − k_2·C_C(t) − 2·k_3·C_C(t)^2
      + 2·k_4·C_D(t) :
```

> de2 := $\frac{d}{dt} C_D(t) = k_3 \cdot C_C(t)^2 - k_4 \cdot C_D(t)$:

> ic := $C_C(0) = 0, C_D(0) = 0$:

Sum of Square of Errors

> res := dsolve({de1, de2, ic}, parameters = $[k_1, k_2, k_3, k_4]$, numeric)

res := **proc**(x_rkf45) ... **end proc** **(4.1)**

> sse := **proc**(k_1, k_2, k_3, k_4)

res(parameters = $[k_1, k_2, k_3, k_4]$) :

add((C_exp[i] - rhs(select(has, res(times[i]), C_C)[ ]))^2 + (D_exp[i] - rhs(select(has, res(times[i]), C_D)[ ]))^2, i = 1 .. numelems(times))

end proc:

> sse(0.01, 0.002, 0.02, 0.002)

2.00578059621453 **(4.2)**

Minimize the Sum of the Square of the Errors

> optPars := Minimize('sse'(k_1, k_2, k_3, k_4), initialpoint = $\{k_1 = 0.011, k_2 = 0.002, k_3 = 0.02, k_4 = 0.002\}$, assume = nonnegative, optimalitytolerance = 0.00001)

optPars := $[0.231411169620961532, [k_1 = 0.0632294154111440, k_2 = 0.0186953983808340, k_3 = 0.0144413290425876, k_4 = 0.]]$ **(5.1)**

Compare Experimental Results to Model

> res := dsolve({de1, de2, ic}, parameters = $[k_1, k_2, k_3, k_4]$, numeric) :

> res(parameters = [optPars[2][ ]]) :

> p_C := odeplot(res, $[t, C_C(t)]$, t = 0 ..70, color = ColorTools:-Color($\left[\frac{149}{255}, \frac{165}{255}, \frac{166}{255}\right]$), legend = "C", filled = true) :

> p_C_exp := plot(times, C_exp, style = point, symbol = solidcircle, symbolsize = 20, color = ColorTools:-Color($\left[\frac{149}{255}, \frac{165}{255}, \frac{166}{255}\right]$)) :

> p_D := odeplot(res, $[t, C_D(t)]$, t = 0 ..70, color = ColorTools:-Color($\left[\frac{58}{255}, \frac{83}{255}, \frac{155}{255}\right]$), legend

```
= "D" ) :
> p_D_exp := plot( times, D_exp, style = point, symbol = solidcircle, symbolsize = 20, color = ColorTools:-
  Color( [ 58/255, 83/255, 155/255 ] ) ) :
> display( p_C, p_C_exp, p_D, p_D_exp, symbol = solidcircle, size = [800, 400], axesfont = [Calibri],
  legendstyle = [font = [Calibri]], labels = ["Time", "Concentration"], labelfont = [Calibri],
  labeldirections = [horizontal, vertical], gridlines )
```