A simulation of the Belousov-Zhabotinsky (BZ) reaction in a Continuous-flow Stirred-Tank

Reactor (CSTR) has been built and investigated. The model for the system is from a paper by

Gyorgyi & Field (1992). Using mainly packages from the Enthought Python Distribution (EPD), the

governing ODEs have been numerically integrated and visualized. The simulator was calibrated by

solving a similar by simpler (non-chaotic) system of ODEs known as the Oregonator- a task

performed successfully. The chaotic system was then investigated under two sets of conditions: a

‘High’ flowrate and a ‘Low’ flowrate, where the flowrate is associated with a set of parameters and

with a particular range of the bifurcation parameter kf, which is defined as an inverse residence time

(s-1) of a particle of fluid in the reactor. All the constants used were from the literature. Although

there were some discrepancies in the results with the literature under the Low flowrate conditions,

there seemed to be better agreement under the High flowrate conditions. Under the Low flowrate

conditions, a period-doubling transition to chaos was observed. The corresponding strange attractor

was virtually 2D. Under the High flowrate conditions, an intermittency route to chaos was observed,

and a 3D attractor was found.

In order to investigate and understand the behavior of the BZ reaction in a CSTR, a Python

(Version 2.5.2) simulation was constructed. ActiveState ® Komodo IDE 5.1.3 was used for the

editing. In general, most of the Python packages were installed with the open-source Enthought

Python Distribution (EPD). Before simulating the more complicated chaotic model given by

Equations 7-9, the simpler model given by Equations 4-6 was simulated as a baseline check on the

overall methodology. Although Equations 4-6 are relatively simple, they are still known to be stiff

ODEs and therefore they require more elaborate integration schemes than, for example, a standard

Runge-Kutta scheme (Epstein and Pojman, 1998). So an ODE integration package from SciPy,

which incorporates variable step-sizes for stiff ODEs, was used. After Equations 4-6 were

successfully integrated and plotted using SciPy’s odeint module, it was assumed that the

methodology was sound and then the same procedure was applied to the chaotic CSTR model,

given by Equations 7-9. The ODE solving routines in odeint are based on popular set of Fortran

ODE solvers called odepack. The vector solution is returned as an array object, and the solutions for

each variable can then be plotted in 2D using matplotlib or in 3D using MayaVi.

There were two categories of flowrate examined for interesting behavior: a ‘Hi’ flow rate and a

‘Low’ flowrate, each corresponding to a particular set of constants, initial conditions, and range of

kf values. For the ‘Low’ flowrate conditions, A = 0.1M, M = 0.25M, H = 0.26M, C = 0.000833M, α

= 666.7, β = 0.3478. For the ‘High’ flowrate conditions, A = 0.14M, M = 0.3M, H = 0.26M, C =

0.001M, α = 333.3, β = 0.2609. The initial conditions were typically based on the points contained

by the Poincaré plane from Gyorgyi and Field (1992).