Proceedings of the Indian National Science Academy (2022) 88:243–249
https://doi.org/10.1007/s43538-022-00081-6

RESEARCH PAPER

Belousov–Zhabotinsky reaction: an open-source approach

Ishaan Misra[1] · V. Ramanathan[1]

Received: 24 March 2022 / Accepted: 3 June 2022 / Published online: 27 June 2022

Abstract
The time-dependent change in concentration of reactants in the oscillatory Belousov–Zhabotinsky reaction defined by non-linear ordinary coupled differential equations has been derived by applying normalized non-dimensional computational methods and presented graphically. The accurate normalisation procedure eschews complex mathematical steps; however, choosing the appropriate references to define the dimensionless variables necessitates a deep physical understanding of the problem. The dimensionless groups are then derived to provide the functional dependences of the unknowns of interest. Finally, to cross-check all the mentioned dependences, a computational simulation has been carried out using domain-specific to general-purpose programming languages. Although there are countless programming languages present to carry out computer simulations, a flexible, portable, and open-source language like python proves fruitful.

Keywords Oscillatory Belousov–Zhabotinsky reaction · Non-dimensional · Computational simulation · Domain-specific · General purpose · Open-source · Python

Introduction

Boris Belousov discovered in the early 1950s that a solution of citric acid, potassium bromate, and cerium (IV) sulphate in dilute sulfuric acid did not react directly to equilibrium, instead turned colorless for a long time before returning to yellow. It is interesting to note that the journals to which this reaction was reported had declined to publish as it was apparently seen to contradict the second law of thermodynamics (Belousov 1959). Later, Anatol Zhabotinsky, a graduate student working under Professor Schnoll's supervision, investigated the reaction in-depth and reported the findings (Zhabotinsky 1964). Zhabotinsky et al. demonstrated that the oscillations in the solution colour were caused by oscillations in the concentration of Ce^{4+} by replacing citric acid with malonic acid. This complex chemical system, popularly referred to as the BZ reaction, as well as related systems, has sparked a lot of interest since then (Field et al. 1972; Field and Noyes 1974; Györgyi et al. 1990; Turanui et al. 1993; Zhabotinsky et al. 1993; Zhabotinsky 2007). This reaction not only has great importance from the physical and chemical point of view, as well as mathematical, but it is also extremely suggestive from a biological point of view (Sánchez-Pérez et al. 2020).

Analogous to the simple pendulum, the oscillations in these reactions are caused by a decrease in potential energy. The energy landscape of a complex reaction is not as smooth as a textbook diagram would suggest. A slow energy-releasing reaction is always present behind all chemical oscillations. The cycle is typically made up of the production and consumption of specific intermediates, which are triggered in order by a species' threshold concentration. Alfred Lotka proposed a mathematical model of this type of predator–prey behaviour in 1910 to describe autocatalytic reactions called the Lotka–Volterra equations (Lotka 1910). These coupled differential equations have been extensively used to describe positive-feedback systems involving two and three variables. Even though they are potential candidates for describing the BZ reactions, there has not been much work in this area, possibly due to the complexity of the system. The overall BZ reaction equation is:

$$3\mathrm{CH_2(COOH)_2} + 4\mathrm{BrO_3^-} + 3C \rightarrow 4\mathrm{Br^-} + 9\mathrm{CO_2} + 6\mathrm{H_2O}$$

Cerium (or manganese) acts as a catalyst in this case. Zhabotinsky discovered that the oxidation of Ce^{3+} by $HBrO_3$ is autocatalytic, and those self-sustained oscillations in Ce^{4+} concentration occur after bromomalonic

✉ V. Ramanathan
vraman.chy@iitbhu.ac.in

[1] Department of Chemistry, Indian Institute of Technology (BHU), Varanasi, India

acid accumulates during the induction period (Zhabotinsky 1964, 2007; Zhabotinsky et al. 1993). A detailed thermodynamic and kinetic analysis was performed by Field, Körös, and Noyes (FKN) to suggest a mechanism for these oscillations (Field et al. 1972; Field and Noyes 1974). Györgi, Turanyi, and Field introduced a mechanistic model for the BZ system with 80 reactions and 26 species (Györgyi et al. 1990). According to the FKN reactions, the 10 processes that define the kinetics of the BZ reactions are (Field et al. 1972):

$$BrO_3^- + 2H^+ + Br^- \rightleftharpoons HBrO_2 + HOBr \tag{1}$$

$$HBrO_2 + H^+ + Br^- \rightleftharpoons 2HOBr \tag{2}$$

$$HOBr + H^+ + Br^- \rightleftharpoons Br_2 + H_2O \tag{3}$$

$$BrO_3^- + HBrO_2 + H^+ \rightleftharpoons 2BrO_2^\circ + H_2O \tag{4}$$

$$BrO_2^\circ + Ce^{3+} + H^+ \rightleftharpoons HBrO_2 + Ce^{4+} \tag{5}$$

$$2HBrO_2 \rightleftharpoons BrO_3^- + HOBr + H^+ \tag{6}$$

$$Br_2 + CH_2(COOH)_2 \rightarrow BrCH(COOH)_2 + H^+ + Br^- \tag{7}$$

$$6Ce^{4+} + CH_2(COOH)_2 + 2H_2O \rightarrow 6Ce^{3+} + HCOOH + 2CO_2 + 6H^+ \tag{8}$$

$$\begin{aligned} 4Ce^{4+} &+ BrCH(COOH)_2 + 2H_2O \rightarrow 4Ce^{3+} \\ &+ HCOOH + Br^- + 2CO_2 + 5H^+ \end{aligned} \tag{9}$$

$$Br_2 + HCOOH \rightarrow 2Br^- + CO_2 + 2H^+ \tag{10}$$

Notwithstanding the cautions of applying Occam's Razor while deducing mechanisms of chemical reactions as articulated by Hoffman et al. (International Journal for Philosophy of Chemistry) we proceed further with the simplistic FKN model also known as Oregonator (origin from the University of Oregon) is the simplest qualitative model to describe BZ reactions(Field and Noyes 1974; Field 2007). It uses steady-state approximation on certain reaction components, particularly BrO_3^-, HOBr, $BrCH(COOH)_2$, Ce^{3+}, H^+andH_2O which do not change during the period of a few cycles. In contrast to this, the concentrations of $HBrO_2$, Br^-andCe^{4+} change drastically over the course of reactions, and thus, their reactions are of significant importance for the overall kinetics of the reaction. The five coupled stoichiometries make up the Oregonator model are:

Step	**Reaction**	**Rate Constant**
(I)	$A + Y \rightarrow X + P$	$k_1 = k_{R1}[H^+]^2$
(II)	$X + Y \rightarrow 2P$	$k_2 = k_{R2}[H^+]$
(III)	$A + X \rightarrow 2X + 2Z$	$k_3 = k_{R4}[H^+]$
(IV)	$2X \rightarrow A + P$	$k_4 = k_{R6}[H^+]$
(V)	$B + Z \rightarrow \frac{1}{2}fZ$	k_5

The species involved in the above reaction scheme are $X = HBrO_2, Y = Br^-, Z = Ce^{4+}, A = BrO_3^-, B = CH_2(COOH)_2$ and$P = HOBr(orBrCH(COOH)_2)$ with adjustable stoichiometric factor f and an adjustable-rate parameter k_5.

In a homogeneous (well-stirred) system, the Law of Mass Action yields a system of three nonlinear ordinary differential rate equations for X, Y, and Z (Field 2007). To present them in a more dimensionless form:

$$\in \frac{dx}{d\tau} = qy - xy + x(1 - x) \tag{i}$$

$$\delta \frac{dy}{d\tau} = -qy - xy + fz \tag{ii}$$

$$\frac{dz}{d\tau} = x - z \tag{iii}$$

by change of variables $x = X/X_o$, $y = Y/Y_o$, $z = Z/Z_o$, $\tau = t/T$ where

$$\begin{aligned} &X_o = \frac{k_3A}{2k_4}, Y_o = \frac{k_3A}{k_2}, Z_o = \frac{(k_3A)^2}{k_4k_5B}, T = \frac{1}{k_5B}, \\ &\in = \frac{k_5B}{k_3A}, \delta = \frac{2k_4k_5B}{k_2k_3A}, q = \frac{2k_1k_4}{k_2k_3} \end{aligned} \tag{14}$$

By using the normalized nondimensionalization, comprising of simple mathematical manipulations (Buckingham 1914; Langhaar 1951; Alhama and Madrid 2012), the dimensionless groups are derived the kinetic differential equations can be reduced from which.

The MATLAB platform offers inbuilt codes to understand this BZ reaction where it digitally analyses a mobile video recording and extracts 24-bit RGB components, and displays the concentration profiles of various chemicals in the BZ reaction, thereby performing a quantitative analysis of experimental data. The Oregonator model's theoretical prediction can be compared to experimental data using MATLAB functions and a standard Runge Kutta routine to solve the system of rate equations (Lozano-Parada et al. 2018).

In this work we provide an alternate way for quantitatively analysing the BZ reaction using the Python platform. There have been efforts made to promote the use of open data and open source computational methods, particularly in the field of chemistry (Guha et al. 2006; O'Boyle et al. 2011). This paper focuses on producing open source (readable and editable) codes in python for simulating and observing the concentrations of the components involved in BZ reaction with time.

Method

For the equations defined above, two methods/techniques are discussed, namely: Runge–Kutta algorithm (4th order) and the Livermore solver for ordinary differential equation (called the LSODE algorithm). A variation of the LSODE algorithm, namely the LSODA algorithm, is used where the suffix A stands for automatic. Although there are other methods for solving non-linear differential equations, such as Heun's method and Euler's method, it has been experimentally proven that the error produced by Runge–Kutta method is 0.625 times that of Heun's method and 0.0007 times that of Euler's method for 32-function evaluations using Euler method, 16-function evaluations using Heun's method, and 8-function evaluations using Runge–Kutta method (https://ece.uwaterloo.ca). Similarly, based on the relative and absolute error bounds supplied in the function, the LSODA algorithm shifts between Adam-Bashford technique (AB) and Backward differentiation method (BDF). Both the AB technique and the BDF are high-precision solvers that are controlled within error limitations in this scenario to provide additional precision and accuracy (Hindmarsh and ODEPACK 1983).

Results and discussion

The codes (shown in the annexure 1) developed by us produce the graphs shown in Fig. 1 using online Jupyter Notebook (https://jupyter.org; https://jupyter.org/try) as the code editor, scipy.integrate (Library for implementation of scientific equations)) as the base function with the initial values of

$$X_o = HBrO_2 = 1, Y_o = Br^- = 1, Z_o = Ce^{4+} = 1 \quad (a)$$

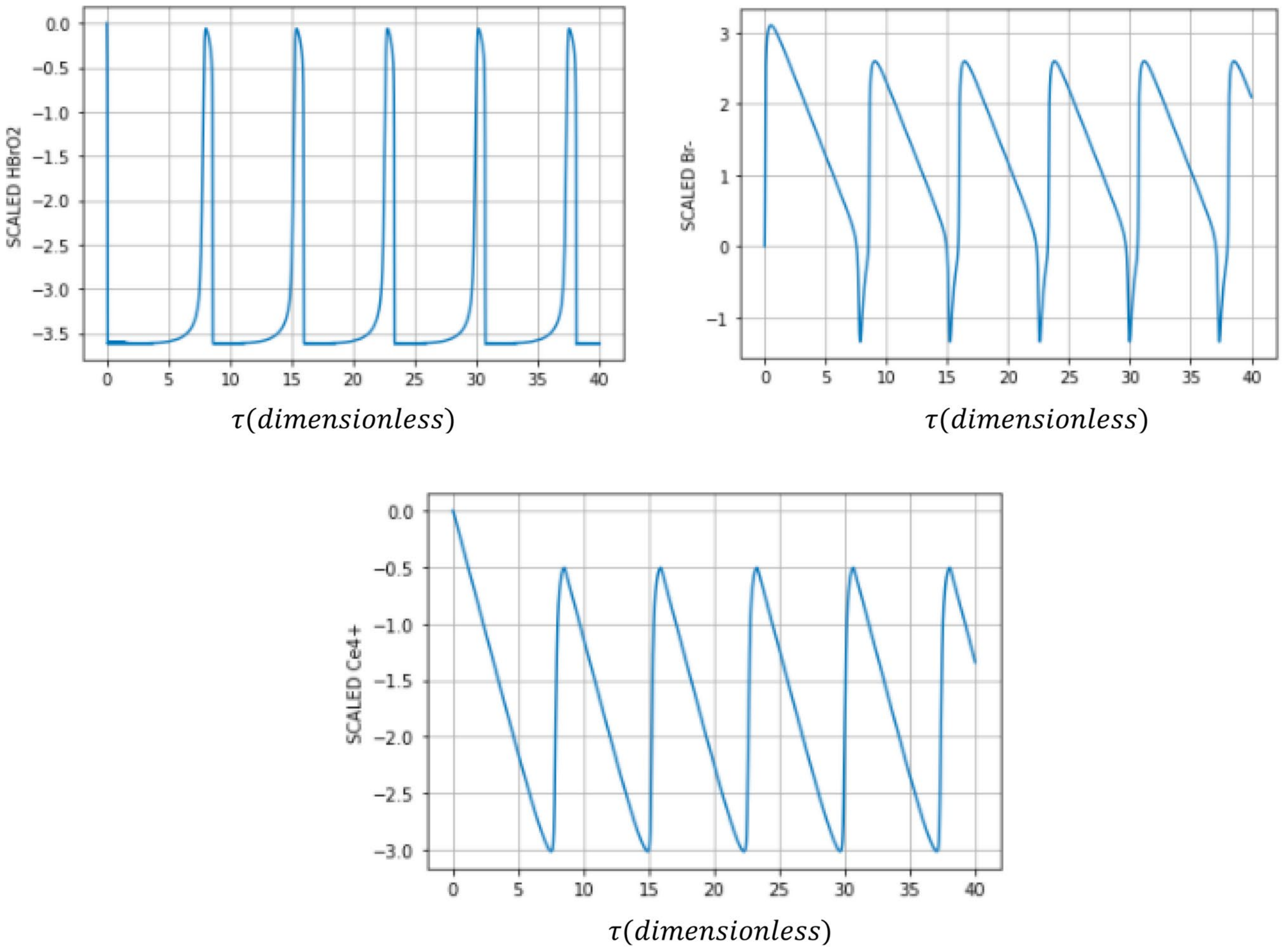

Fig. 1 Identical results are obtained from both the algorithms namely RK45 and LSODA

and the dimensionless parameters for the Eqs. (i), (ii) and (iii) being $\epsilon = 3.6 \times 10^{-2}$, $\delta = 1.2 \times 10^{-4}$ and $q = 2.4 \times 10^{-4}$ (Barzykina 2020). These dimensionless parameters are derived from the values $k_{R1} = 2$ $mol^{-3} \cdot dm^{9}$ s^{-1}, $k_{R2} = 1.8 \times 10^{6}$ $mol^{-2} \cdot dm^{6}$ s^{-1}, $k_{R4} = 48$ $mol^{-2} \cdot dm^{6}$ s^{-1}, $k_{R6} = 3 \times 10^{3}$ $mol^{-1} \cdot dm^{3}$ s^{-1}, $[H^{+}] = 1.68$ $mol \cdot dm^{-3}$, $A = [BrO_3{}^{-}] = 0.058$ $mol \cdot dm^{-3}$, and $B = [CH_2(COOH)_2] = 0.105$ $mol \cdot dm^{-3}$, along with $k_5 = 1.6$ $mol^{-1} \cdot dm^{3}$ s^{-1}, $f = 1$ as proposed by Fields (2007), and the Eqs. *(I) to (iv)* shown above.

Here scaling the values signifies,

$$\mathrm{ScaledHBrO_2} = \log_{10}\left(\frac{X}{X_o}\right), \mathrm{ScaledBr^-} = \log_{10}\left(\frac{Y}{Y_o}\right), \mathrm{ScaledCe^{4+}} = \log_{10}\left(\frac{Z}{Z_o}\right)$$

When numerically integrating a differential equation, it is expected for the required step size to be small in areas where the solution curve exhibits a lot of variation and large in areas where the solution curve straightens out to approach a line with a slope near zero. In some cases, a numerical method must have an unacceptably small step size in a region where the solution curve is very smooth in order to provide a reliable solution to the differential system. This phenomenon is called stiffness. The equations which follow this phenomenon or property are called stiff differential equations and vice versa. There are a few numerical methods that are more commonly used to solve these types of differential equations (Hindmarsh and "ODEPACK 1983). Runge–Kutta integration method used here (called RK45) is an explicit method or order 5(4) (Dormand and Prince 1980). The error is controlled based on the fourth-order method's accuracy, but steps are taken based on the fifth-order precise formula (with local extrapolation). For the dense output, a quartic interpolation polynomial is used (Shampine 1986). In general, it can be applied in the complex domain to non-stiff differential equation problems. Ordinary Differential Equation Solver for Stiff or Non-Stiff Systems, on the other hand, is LSODA. This is a wrapper of the Fortran solver from ODEPACK (Barzykina 2020). When the problem is stiff, LSODA uses a dense or banded Jacobian to solve the system dy/dt = f, but it automatically chooses between non-stiff (Adams) and stiff (BDF) methods (Petzold 1980). It starts with the non-stiff method and then monitors data in real time to decide which method to use. To make modification easier, the LSODA source is heavily commented. There are two versions available: a single-precision version and a double-precision version.

It was possible to achieve a remarkable agreement of the theoretically predicted oscillation curve in MATLAB with a theory for cerium(IV) as a catalyst by taking the rate constants of the processes involved from the literature and substituting the conditions of the current study to the results obtained above (Barzykina 2020).

The RK45 and LSODA codes can be compared based on run-time and the preciseness (number of iterations) of both programs. The run time of a program can be understood as the total time required to perform the given set of operations in the same computer system. In this case, we take the start of the run-time before initializing the dimensionless parameters and end the same after plotting the curve for both algorithms. The number of iterations defines preciseness. An increase in the number of iterations or the number of data points within the given time interval (in this case [0,40]) produced will, in turn, reduce the scope of the error in the graphs produced.

Using the timeit module (https://docs.python.org) in python, the performance of both the algorithms executed on the computer machine with a 1.8 GHz processor is compared in Table 1.

From the above table, it is evident that the run time of RK45 algorithm is almost 22 times that of the LSODA algorithm which will remain same for all processing systems. From the result obtained, it can be observed that the RK45 algorithm creates many data points in the same time interval at the cost of the run time of the overall program.

Although both the codes work successfully, the code using the LSODA algorithm can certainly be extended to problems with bigger datasets for faster calculation. The RK45 algorithm, on the other hand, can certainly be used for problems involving high precision and accuracy since it produces large datasets in the same time frame.

There is a shift in $HBrO_2$ concentration in a poorly stirred BZ reaction system. It functions as a limiter at each stage, limiting the amount of $HBrO_2$ that can be used in that iteration. This indicates that if the $HBrO_2$ concentration obtained in the previous iteration is higher than the one necessary in the next, the maximum $HBrO_2$ concentration will be used in the computation. Because the code is iterative, running it with a simple conditional statement to see if the limit concentration of $HBrO_2$ is equal to the current concentration will provide the desired graph (https://ro.uow.edu.au).

Table 1 Comparison of RK45 and LSODA algorithms

	Run time	Number of iterations
LSODA algorithm	0.31 s	1972
RK45 algorithm	6.85 s	48,283

Conclusion

The Field, Körös, and Noyes (FKN) model for the mechanism of the BZ reaction, which involves three cyclic stages: bromate reduction by bromide, autocatalytic production of bromous acid accompanied by oxidation and reduction of the metal ion catalyst, and regeneration of bromide ion, was taken up in this work and the corresponding dimensionless differential equations were successfully solved using a novel open-source approach.

It is hereby conclusively demonstrated that the simulation models of the BZ reaction from a computing language like MATLAB can be extended to a more futuristic general-purpose language like python. Two methods using RK45 and the LSODA algorithm respectively were used to produce the same result as obtained with a MATLAB ODE45 code.

Extension of the current study would be rewarding because BZ reactions are fascinating and rich in their displayed behavior. These pieces of code are general in nature. With a little tweak in the structures of differential equations and the parameters involved graph for the desired oscillatory system can be produced. Another interesting project that can be derived from this paper is to generalize this piece of code for non-oscillatory systems to observe various reactions.

Annexure 1

Code using RK45 algorithm

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.integrate as integ
# Dimensionless parameters
ef=0.00012
e=0.036
f=1
q=0.00024
#packing 3 initial conditions with state of x,y,z into y0
initial_values = [1,1,1]
#Differential equation to be computed wrapped in a function
def Calculate (t,Y):
  x = Y[0]
  y = Y[1]
  z = Y[2]
  dxdt = (q*y - x*y+ x- x*x)*(1/e)
  dydt = (-1*q*y-x*y+f*z)*(1/ef)
  dzdt = x-z
  return [dxdt,dydt,dzdt]
#Computing the values
solution=integ.solve_ivp(Calculate,[0, 40],initial_values,method='RK45', atol=1e-4, rtol=1e-6)
Y=solution.y
timee=solution.t
#Scaling the result obtained for better visualization
xvalnew,yvalnew,zvalnew=[],[],[]
tempp=0
while tempp<len(Y[0]):
    xvalnew.append(np.log10(Y[0][tempp]))
    yvalnew.append(np.log10(Y[1][tempp]))
    zvalnew.append(np.log10(Y[2][tempp]))
    tempp+=1
#Plotting the graph
plt.plot(timee,xvalnew)        #Plotting HBrO2
plt.plot(timee,yvalnew)        #Plotting Br-
plt.plot(timee,zvalnew)        #Plotting Ce4+
plt.grid()
plt.show()
```

Code using LSODA algorithm

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.integrate as integ
# Dimensionless parameters
ef=0.00012
e=0.036
f=1
q=0.00024
#packing 3 initial conditions with state of x,y,z into y0
initial_values = [1,1,1]
#Differential equation to be computed wrapped in a function
def Calculate (t,Y):
  x = Y[0]
  y = Y[1]
  z = Y[2]
  dxdt = (q*y - x*y+ x- x*x)*(1/e)
  dydt = (-1*q*y-x*y+f*z)*(1/ef)
  dzdt = x-z
  return [dxdt,dydt,dzdt]
#Computing the values
solution=integ.solve_ivp(Calculate,[0, 40],initial_values,method='LSODA', atol=1e-4, rtol=1e-6)
Y=solution.y
timee=solution.t
#Scaling the result obtained for better visualization
xvalnew,yvalnew,zvalnew=[],[],[]
tempp=0
while tempp<len(Y[0]):
    xvalnew.append(np.log10(Y[0][tempp]))
    yvalnew.append(np.log10(Y[1][tempp]))
    zvalnew.append(np.log10(Y[2][tempp]))
    tempp+=1
#Plotting the graph
plt.plot(timee,xvalnew)        #Plotting HBrO2
plt.plot(timee,yvalnew)        #Plotting Br-
plt.plot(timee,zvalnew)        #Plotting Ce4+
plt.grid()
plt.show()
```

*These codes can also be publicly accessed in this github directory: https://github.com/ishaanms/METHODS_FOR_SOLVING_BZ_N.git

#reviewer1

Citations made in order

#reviewer 3

1. MAtlab uses ode23 I use ode45 meaning more precision

   MATLAB CODE::
2. https://ro.uow.edu.au/cgi/viewcontent.cgi?article=1202c&context=eispapers

   Acc to the paper for a poorly stirred reaction the conc of hbro2 will vary to give a bifurcation curve

   In my case it can be understood that for every data point to be produced we will have hbro2 as limiting reagent which will give (variable max hbro2 conc during the reaction)
3. efficiency and accuracy is defined by rtol and atol used in the code(relative error and absolute error); minimised as much as possible particularly for bz reaction because things do not go this precise
4. runtime reduced to 4 decimal places

Acknowledgements The authors thank the academic infra structure of IIT(BHU) Varanasi.

Author contributions VR conceptualized the idea, IM wrote and executed the python code. IM and VR together wrote the draft.

Declarations

Conflict of interest The authors declare absolutely no conflict of interests.

References

Alhama F, Madrid CN (2012) Análisis Dimensional Discriminado En Mecánica De Fluidos Y Transferencia De Calor. Reverté: Barcelona

Barzykina, I.: Chemistry and mathematics of the BelousovZhabotinsky reaction in a school laboratory. J. Chem. Educ. **97**(7), 1895–1902 (2020)

Belousov, B.P.: Collection of Short Papers on Radiation Medicine for 1958, pp. 145–147. Medgiz, Moscow (1959)

Buckingham, E.: On physically similar systems; illustration of the use of dimensional equations. Phys. Rev. **4**, 345 (1914)

Dormand, J.R., Prince, P.J.: A family of embedded Runge–Kutta formulae. J. Comput. Appl. Math. **6**(1), 19–26 (1980)

Field, R.J.: Noyes RM (1974) Oscillations in chemical systems: IV: limit cycle behavior in a model of a real chemical reaction. J. Chem. Phys. **60**, 1877–1884 (1974)

Field, R.R.: Oregonator. Scholarpedia **2**, 1386 (2007)

Field, R.J., Körös, E., Noyes, R.M.: Oscillations in chemical systems. II: thorough analysis of temporal oscillation in the bromatecerium-malonic acid system. J. Am. Chem. Soc. **94**, 8649–8664 (1972)

Guha, R., Howard, M.T., Hutchison, G.R., Murray-Rust, P., Rzepa, H.S., Steinbeck, C., Wegner, J.K., Willighagen, E.: The blue obelisk interoperability in chemical informatics. J. Chem. Inf. Model. **46**, 991–998 (2006)

Györgyi, L., Turanui, T., Field, R.J.: Mechanistic details of thé oscillatory Belousov-Zhabotinskii reaction. J. Phys. Chem. **94**, 7162–7170 (1990)

Hindmarsh, A. C.: ODEPACK, A Systematized Collection of ODE Solvers. In: Stepleman, R.S. et al. (eds.) Scientific Computing, pp. 55–64. North-Holland, Amsterdam, (vol. 1 of IMACS Transactions on Scientific Computation) (1983)

https://docs.python.org/3/library/timeit.html

https://ece.uwaterloo.ca/~dwharder/NumericalAnalysis/14IVPs/multistep/error.html

https://jupyter.org/

https://jupyter.org/try

https://ro.uow.edu.au/cgi/viewcontent.cgi?article=1202c&context=eispapers

Langhaar, H.L.: Dimensional Analysis and Theory of Models. Wiley, New York (1951)

https://www.scipy.org/ (Library for implementation of scientific equations)

Lotka, A.J.: Contribution to the theory of periodic reactions. J. Phys. Chem. **14**, 271–274 (1910)

Lozano-Parada, J.H., Burnham, H., Machuca Martinez, F.: Pedagogical approach to the modeling and simulation of oscillating chemical systems with modern software: the Brusselator model. J. Chem. Educ. **95**, 758–766 (2018)

O'Boyle, N.M., Guha, R., Willighagen, E.L., et al.: Open data, open source and open standards in chemistry: the blue obelisk five years on. J Cheminform **3**, 37 (2011)

Petzold, L. R.: Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations, Sandia National Laboratories Report SAND80–8230 (1980).

Sánchez-Pérez, J.F., Conesa, M., Alhama, I., Cánovas, M.: Study of Lotka-Volterra biological or chemical Oscillator problem using the normalization technique: prediction of time and concentrations. Mathematics **8**(8), 1324 (2020)

Shampine, L.W.: Some practical Runge–Kutta formulas. Math. Comput. **46**(173), 135–150 (1986)

Turanui, T., Gyorgyi, L., Field, R.J.: Analysis and Simplification of the GTF Model of the Belousov–Zhabotinsky Reaction. J. Phys. Chem. **97**, 1931–1941 (1993)

Zhabotinsky, A.M.: Periodic oxidation of malonic acid in solution (a study of the Belousov reaction kinetics). Biofizika **9**, 306–311 (1964)

Zhabotinsky, A.M.: Belousov–Zhabotinsky reaction. Scholarpedia **2**, 1435 (2007)

Zhabotinsky, A.M., Buchholtz, F., Kiyatkin, A.B., Epstein, I.R.: Oscillations and waves in metal-ion-catalyzed bromate oscillating reactions in highly oxidized states. J. Phys. Chem. **97**, 7578–7584 (1993)