Mathematical support for solving the direct problem of chemical kinetics to clarify the main patterns of pollutant emissions

Aleksandr Terekhov*, Federico* Huaman de los Eros*, Fayruza* Ziganshina*, Svetlana* Mavlyutova*, Ilya* Bespalov*, and *Aydar* Galimov

Ufa State Petroleum Technological University, Ufa, RussianFederation

Abstract. Determination of kinetic parameters and obtaining the dependence of concentrations of substances on time is the main stage in the construction of a kinetic model. The mathematically constructed model is presented in the form of systems of differential equations. At this stage, they usually deal with two types of tasks, which are called forward and reverse. This paper presents algorithms for solving an arbitrary direct problem of chemical kinetics. As a method for solving systems of differential equations, the Rosenbrock type method is considered. The advantage of this work is to describe the solution of the direct problem for an arbitrary kinetic scheme. The input data for the program is a kinetic scheme, the output is a constructed mathematical model and the found parameters. The purpose of creating a program: finding parameters, that is, solving arbitrary found systems of differential equations based on kinetic schemes. This article describes the construction of algorithms for solving systems of homogeneous differential equations based on the methods described above.

1 Introduction

In mathematical modeling of chemical processes, they face such problems as:

1. Determination of kinetic concentrations of substances involved in the process, i.e. solution of the direct problem. Solving the direct problem boils down to finding kinetic curves that represent the dependence of the concentrations of the substances involved in the reaction on time;
2. Finding kinetic parameters, that is, solving the inverse problem. In solving the inverse problem, experimental values of the concentrations of substances, which depend on time, are used to find the unknown on the right side of the equation. The kinetic parameters that determine the mechanism of reactions are unknown.

Solving the direct problem uses a complex reaction scheme or a hypothesized model that reflects the mechanism of the chemical process. In formal kinetics, the concept of mechanism is used as a set of stages through which these reagents are obtained into products. Based on the assumed kinetic scheme, a mathematical reaction model is compiled, which is presented in the form of systems of differential equations.

* Corresponding author: Waaa20111@yandex.ru

Algorithms for solving the direct problem of chemical kinetics using a programming language are divided into stages:

1. Input as chemical reaction kinetic scheme, rate constants, initial concentrations, molecular matrix, integration range and integration step.
2. Compiling a mathematical reaction model in the form of differential equations. Differential equations describe the rate of flow or accumulation of each reaction component.
3. Solutions of systems of differential equations using numerical methods, namely Rosenbrock type methods.

When solving direct problems of chemical kinetics, a situation arises with the simultaneous presence of slow and fast chemical reactions when the reaction rate constants differ by several orders of magnitude, then the ODE system describing the reaction data is rigid. To solve this kind of ODE systems, that is, rigid systems, it is recommended to use the Rosenbrock method.

To solve rigid systems, you can use the Runge Kutta method with an adaptive step, but the tougher the system, the more steps are required, and this leads to a lot of calculations.

Therefore, for such systems, it is more expedient to use the Rosenbrock method and the (4,2) -method [5], which are also applicable to solving systems that are not rigid.

This article describes the construction of algorithms for solving systems of homogeneous differential equations based on the methods described above.

Below is an example of a kinetic scheme and detailed algorithms for implementing three stages that are implemented in the Delphi programming language. To implement the second stage - building a mathematical model here, it is necessary to highlight several stages.

1) Construction of a matrix of stoichiometric coefficients.
2) Preparation of reaction velocity vector
3) Output of the system of differential equations to the program interface.
4) Calculation of the values of the system of differential equations at the required point in the interval.

The algorithm for constructing a matrix of stoichiometric coefficients consists in counting each reacting substance and product in each stage, if any reagent or component that does not participate in the considered stage, then the corresponding stoichiometric coefficient is zero. The stoichiometric coefficient of reagents is marked with a minus sign, for products - positive.

2 Materials and methods

At the first stage of solving the direct problem, it is necessary to choose a method for solving differential equations, taking into account the features of the system itself. One such method for solving systems of differential equations is the Rosenbrock-type method. This method is discussed in [1-3].

The mathematical basis for describing the reaction is the law of conservation of mass, molecular matrix, stoichiometric matrix [2,3].

The molecular matrix reflects the chemical composition of the substances. The elements of the matrix are the number of atoms of the j-th element that are included in the molecule of the i-th substance.

The stoichiometric matrix consists of elements that are stoichiometric coefficients of the substances involved in the reactions.

Since all the processes under consideration are in a closed system, it is necessary to use the law of conservation of mass

$$\sum_i m_{i_0} = \sum_i m_i$$

where m (i_0) and m_i are initial and current masses of the i-th substance.

These concepts are discussed in the example below.

Reaction system is given

$$2C+O2=2CO;$$
$$2CO+O2=2CO2;$$
$$C+O2=CO2.$$

The molecular matrix for reagents C, O2, CO, CO2 has the form:

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}$$

The rows in this matrix correspond to substances C, O2, CO, CO2, and the columns correspond to elements C and O.

The stoichiometric matrix B for the reaction system has the form:

$$B = \begin{pmatrix} -2 & -1 & 2 & 0 \\ 0 & -1 & -2 & 2 \\ -1 & -1 & 0 & 1 \end{pmatrix}$$

The law of constancy of mass by stages in matrix form, for the example under consideration, is as follows:

$$BA = 0$$

$$\begin{pmatrix} -2 & -1 & 2 & 0 \\ 0 & -1 & -2 & 2 \\ -1 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} =$$

$$\begin{pmatrix} -2 \cdot 1 + (-1) \cdot 0 + 2 \cdot 1 + 0 \cdot 1 & -2 \cdot 0 + (-1) \cdot 2 + 2 \cdot 1 + 0 \cdot 2 \\ 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + 2 \cdot 1 & 0 \cdot 0 + (-1) \cdot 2 + (-2) \cdot 1 + 2 \cdot 2 \\ -1 \cdot 1 + (-1) \cdot 0 + 0 \cdot 1 + 1 \cdot 1 & -1 \cdot 0 + (-1) \cdot 2 + 0 \cdot 1 + 1 \cdot 2 \end{pmatrix}$$

When solving direct problems, mathematical software should include the construction of a system of differential equations, the corresponding reaction mechanism, the stoichiometric matrix, the law of conservation of mass, the molecular matrix.

In works [4], the construction of a kinetic model and particular solutions of the forward and backward problems are considered.

The advantage of this work is to describe the solution of the direct problem for an arbitrary kinetic scheme. The input data for the program is a kinetic scheme, the output is a constructed mathematical model and the found parameters.

To implement the first stage, it is necessary to solve the issue of entering initial data, i.e. creating an understandable interface for the end user.

As an example, Table 1 shows the kinetic scheme of the chemical reaction.

Table 1. Example reaction mechanism.

№	Stage number	Reaction	Kinetic parameter values
1	1-2	$X_1 \Leftrightarrow X_2$	k_1=15; k_2=179
2	3	$X_1 \rightarrow P_1$	k_3=305
3	4	$X_1 \rightarrow X_3$	k_4=9
4	5	$X_2 \rightarrow X_3$	k_5=16
5	6	$X_2 + X_3 \rightarrow D$	k_6=6·105
6	7	$D \rightarrow X_2 + X_2$	k_7=3
7	8	$D \rightarrow P_2$	k_8=106

As you can see, the rate constants vary greatly, which indicates the rigidity of the system of differential equations.

The kinetic scheme of the chemical reaction and the rate constants are recorded in the program table in the form of sequential reactions

Table 2. Kinetic circuit input table SG.

№	Reaction	Kinetic parameter values
1	$X_1 = X_2$	k_1=15
2	$X_2 = X_1$	k_2=179
3	$X_1 = P_1$	k_3=305
4	$X_1 = X_3$	k_4=9
5	$X_2 = X_3$	k_5=16;
6	$X_2 + X_3 = D$	$k_6=6\cdot10^5$;
7	$D = X_2 + X_2$	k_7=3;
8	$D = P_2$	$k_8=10^6$;

Information from this table is transmitted to a two-dimensional array ST, with reactions written in st [1, i], and rate constants in st [2, i], where i = 1.. n, n is the number of stages, i.e.

for i:=1 to n do begin

st[1,i]:=SG.cells[1, i];

st[2,i]:=SG.cells[2, i];

Initial concentrations are entered into the table SG2

Table 3. Table SG2 entry of initial concentrations.

№ п/п	Reaction	Initial Concentrations
1	X1	0.00000243
2	X2	0.0000020224
3	P1	0
4	X3	0
5	D	0
6	P2	0

Values of initial concentrations are recorded in array A [j], where j = 1.. m, m - quantity of reagents. For this example, m = 6.

The interval and integration step are entered via the menu Direct task → Initial data.

The interval and integration step boundaries are written to variables c, d, h, respectively.

Using the procedure reagentProduct (st, s1) in the created program, reaction reagents are transferred to the variable S1 in the form of text. The algorithm of this procedure is to search for reagents included in each reaction. All found reagents are written to the variable S, for

the example under consideration, the variable S consists of elements X1, X2, X2, X1, X1, P1, X1, X3, X2, X3, X2, X3, D, D, X2, X2, D, P2. Next, you need to remove duplicate items. As a result, the content of the variable S1 consists of the following list of reagents and products: X1, X2, P1, X3, D, P2.

After reading the initial data, the program builds a mathematical model. This program solves direct problems of chemical kinetics for closed systems. The kinetic model of a complex reaction in a closed system is presented:

$$f = B^T sp(i),$$

where f - vector - column of substance concentration changes; BT - transposed stoichiometric matrix; sp (i) is the stage velocity column vector.

3 Results and discussion

The algorithm for constructing a matrix of stoichiometric coefficients consists in counting each reacting substance and product in each stage, if any reagent or component that does not participate in the considered stage, then the corresponding stoichiometric coefficient is zero. The stoichiometric coefficient of reagents is marked with a minus sign, for products - positive.

For the example under consideration, this matrix looks like this

	X1	X2	P1	X3	D	P2
1	-1	1	0	0	0	0
2	1	-1	0	0	0	0
3	-1	0	1	0	0	0
4	-1	0	0	1	0	0
5	0	-1	0	1	0	0
6	0	-1	0	-1	1	0
7	0	2	0	0	-1	0
8	0	0	0	0	-1	1

Fig. 1. Stoichiometric coefficient.

The stoichiometric coefficient values are stored in a two-dimensional array B [i, j]. After obtaining the matrix of stoichiometric coefficients, the transpose of this matrix is performed.

The algorithm for obtaining the reaction velocity vector consists in considering the left side of each stage and replacing the "+" sign with the "*" sign, then the speed constant is assigned to the front of the resulting record. It should be noted that the resulting rates for each stage are stored as text in the sp variable. The content of the sp variable for this example is as follows:

$$sp[i] = \begin{bmatrix} k_1 * X_1 \\ k_2 * X_2 \\ k_3 * X_1 \\ k_4 * X_1 \\ k_5 * X_2 \\ k_6 * X_2 * X_3 \\ k_7 * D \\ k_8 * D \end{bmatrix}$$

The sp variable is needed to output differential equations to the SG3 interface table of this program and to calculate the numerical values of the speeds of each stage.

Use the output procedure to output a system of differential equations to the program interface SG3 table.

Construction of differential equation system is performed by multiplication of stoichiometric matrix and velocity vector

$$\begin{bmatrix} -1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} k_1 * X_1 \\ k_2 * X_2 \\ k_3 * X_1 \\ k_4 * X_1 \\ k_5 * X_2 \\ k_6 * X_2 * X_3 \\ k_7 * D \\ k_8 * D \end{bmatrix} \quad (1)$$

then, for the example in question, the system of differential equations is as follows:

$$\begin{aligned} \frac{d[X_1]}{dt} &= -k_1[X_1] + k_2[X_2] - k_3[X_1] - k_4[X_1] \\ \frac{d[X_2]}{dt} &= k_1[X_1] - k_2[X_2] - k_5[X_2] - k_6[X_2][X_3] + 2k_7[D] \\ \frac{d[X_3]}{dt} &= k_4[X_1] + k_5[X_2] - k_6[X_2][X_3] \\ \frac{d[D]}{dt} &= k_6[X_2][X_3] - k_7[D] - k_8[D] \\ \frac{d[P_1]}{dt} &= k_3[X_1] \\ \frac{d[P_2]}{dt} &= k_8[D] \end{aligned} \quad (2)$$

Obtaining numerical values of a system of differential equations is carried out using the function f (l: integer; k, a1: mas5) and du (a1: mas5; var w: mas5).

At the required point in the integration interval, the program accesses the function f (l: integer; k, a1: mas5), the number of the differential equation I, the rate constants k and the values of the current concentrations of substances a1 are transmitted to this function. In the function in question, the du (a1: mas5; var w: mas5) to calculate the stage velocity vector w. Using the number of the differential equation I, the matrix of stoichiometric coefficients B [l, i] and the velocity vector of the stages w, the function f determines the value of the necessary differential equation. The differential equation number determines which row to use from the stoichiometric matrix to calculate the value of the desired differential equation.

For example, consider the calculation of the second differential equation, then you need to multiply the second row of the stoichiometric matrix by the velocity vector

$$[1 \quad -1 \quad 0 \quad 0 \quad -1 \quad -1 \quad 2 \quad 0] * \begin{bmatrix} k_1 * X_1 \\ k_2 * X_2 \\ k_3 * X_1 \\ k_4 * X_1 \\ k_5 * X_2 \\ k_6 * X_2 * X_3 \\ k_7 * D \\ k_8 * D \end{bmatrix} =$$

$$k_1[X_1] - k_2[X_2] - k_5[X_2] - k_6[X_2][X_3] + 2k_7[D] \quad (3)$$

As a method for solving the system, Rosenbrock-type methods are considered. The algorithm of the Rosenbrock type method is to calculate the following formulas:

$$\begin{aligned} y_{n+1} &= y_n + \sum_{i=1}^{m} p_i\, k_i, \\ D_{n,k} k_i &= hf\left(y_n + \textstyle\sum_{j=1}^{i-1} \beta_{ij}\, k_j\right), \\ D_{n,i} &= E - a_i hJ\left(y_n + \textstyle\sum_{j=1}^{i-1} \gamma_{ij}\, k_j\right), \end{aligned} \quad (4)$$

where h — integration step; E - the identity matrix; J =∂f(yn)/∂y — Jacobian matrix of vector function f(y); ai, pi, βij and γij, $1 \le i \le m$, $1 \le j \le i - 1$, — numerical coefficients determining properties of accuracy and stability. Currently, Rosenbrock-type methods are presented in detail in [6].

In this work, a two-stage Rosenbrock method of the third order of accuracy [7] was used to solve systems of differential equations:

$$y\_(n+1)=y\_n+p\_1\ z\_1+p\_2\ z\_2, \quad (5)$$

$$D_{n,1} = E - ha_1 J(y_n), \quad (6)$$

$$z_1 = h\left[D_{n,1}\right]^{-1} f(y_n) \quad (7)$$

$$D_{n,2} = E - ha_2 J(y_n + \gamma_1 k_1), \quad (8)$$

$$z_2 = h[D_{n,2}]^{-1} f(y_n + \beta_1 k_1) \quad (9)$$

$$a_1 = 1 + \sqrt{6}/6 = 1.40824829046386;$$

$$a_2 = 1 - \sqrt{6}/6 = 5.91751709536137e - 1;$$

$$p_1 = -0.41315432;$$

$$p_2 = 1.41315432;$$

$$\beta_1 = \gamma_1 = \frac{\left(-6 - \sqrt{6} + \left[58 + 20\sqrt{6}\right]^{\frac{1}{2}}\right)}{6 + 2\sqrt{6}} = 1.73786673924946e - 1;$$

1. The implementation of this method consists of four stages. These steps are discussed in detail below.
2. Finding the Jacobi matrix J using the procedure jacob(a, k:mas5;varjcb:mas7), where a – current concentrations of substances, k are rate constants, jcb is the Jacobi matrix found. As you know, the Jacobi matrix is a matrix consisting of all kinds of partial derivatives for a system consisting of m functions. To calculate the Jacobi matrix, it is necessary to find partial derivatives of all functions of the system over all variables. For internal mesh nodes, the central differential derivative is used, and for extreme nodes, the one-sided derivative is used [8].

$$J[i,j] = \frac{f(x_1, \dots x_l + h, \dots, x_m) - f(x_1, \dots x_l - h, \dots, x_m)}{2h}, i = 2 \dots n - 1;$$

$$J[i,j] = \frac{f(x_1, \dots x_l + h, \dots, x_m) - f(x_1, \dots x_l, \dots, x_m)}{h}, i = 1;$$

$$J[i,j] = \frac{f(x_1, \dots x_l, \dots, x_m) - f(x_1, \dots x_l - h, \dots, x_m)}{h}, i = n;$$

3. Finding the matrix Dn, i - of formula 6 or 8 and the inverse matrix (Dn, i) -1, is necessary to calculate the coefficients zi.
4. Calculation of coefficients zi by formulas (7) or (9).
5. The developed procedures and finding yn + 1 according to formula (5) are carried out in the rsnbrk procedure (hh:extended; k:mas5;a:mas5; varanext: mas 5), where hh - the integration step, k - the rate constants, a - the current values of the concentrations of substances or yn, anext are the values of the concentrations of substances at a new point in the integration interval.

4 Conclusion

Algorithms for solving the direct problem of chemical kinetics using a programming language are divided into stages: input as chemical reaction kinetic scheme, rate constants, initial concentrations, molecular matrix, integration range and integration step; compiling a mathematical reaction model in the form of differential equations. Differential equations describe the rate of flow or accumulation of each reaction component; solutions of systems of differential equations using numerical methods, namely Rosenbrock type methods.

To implement the second stage - building a mathematical model here, it is necessary to highlight several stages: construction of a matrix of stoichiometric coefficients; preparation of reaction velocity vector; output of the system of differential equations to the program interface; calculation of the values of the system of differential equations at the required point in the interval.

This work proposes algorithms for compiling a mathematical model for an arbitrary kinetic scheme with the subsequent solution of a direct problem. Based on these algorithms, a program has been created that implements the following stages related to the construction of a kinetic model and the solution of a direct problem.

1. Introduction of kinetic scheme, parameters and data using the developed interface.
2. Obtaining a system of differential equations based on the input kinetic scheme (mathematical model).
3. Development of algorithms for solving a system of differential equations that are based on the Rosenbrock-type method.
4. Obtaining and plotting the dependence of concentrations of reacting substances and products on time.

References

1. E.A. Novikov, Informatics and control systems **1(43)**, 54-62 (2015).
2. A.M. Zubanov, N.I. Kokonkov, P.D. Shirkov, Mathematical modelling **23(3)**, 127-138 (2011).
3. V.I. Korobov, V.F. Ochkov, *Chemical kinetics: introduction with Mathcad, Maple, MCS* (Hotline-Telecom, Moscow, 2009).
4. F.T. Badrtdinova, S.I. Spivak, V.M. Goldberg, L.A. Bigaeva, Plastic masses **7** (2011).
5. M.P. Galanin, S.R. Khojaeva, Mathematical modeling and numerical methods **4**, 95-119 (2014).
6. E. Hairer, G. Wanner, *Solving ordinary differential equations. Rigid and differential algebraic problems* (Mir, Moscow, 1999).
7. A.B. Alshin, E.A. Alshina, A.G. Limonov, J. mathem. and mathem. Physical **49(2)**, 270-287 (2009).
8. V.M. Verzhbitsky, *Fundamentals of numerical methods. Textbook for universities* (Higher. shk., Moscow, 2002).