Solving Stiff Systems of ODEs by Explicit Methods with Conformed Stability Domains

A. E. Novikov
Institute of Mathematics and Fundamental Informatics
Siberian Federal University
Krasnoyarsk, Russia
aenovikov@bk.ru

M. V. Rybkov
Institute of Mathematics and Fundamental Informatics
Siberian Federal University
Krasnoyarsk, Russia
mixailrybkov@yandex.ru

Yu. V. Shornikov
Automation and Computer Engineering Department
Novosibirsk State Technical University
Novosibirsk, Russia
shornikov@inbox.ru

L. V. Knaub
Institute of Mathematics and Fundamental Informatics
Siberian Federal University
Krasnoyarsk, Russia
lvknaub@yandex.ru

Abstract — **The Cauchy problem for a stiff system of ODEs is considered. The explicit m-stage first order methods of the Runge-Kutta type are designed with stability domains of intermediate numerical schemes conformed with the stability domain of the basic scheme. Inequalities for accuracy and stability control are obtained. A numerical algorithm based on the first-order method and the five-stage fourth order Merson method is developed. The algorithm is aimed at solving large-scale systems of ODEs of moderate stiffness with low accuracy. It has been included in the library of solvers of the ISMA simulation environment. Numerical results showing growth of the efficiency are given.**

Keywords - Runge-Kutta methods; accuracy and stability control; conformed stability domains; stiff problems;

I. INTRODUCTION

Nowadays software for mathematical modelling and simulation is widely used for describing different processes in Chemical Kinetics, Electrotechnics and other applications. Models often are defined via either systems of ODEs or systems of PDEs. At that, systems of PDEs can be transformed to systems of ODEs applying discretization spatial derivatives. The greater the discretization step, the higher dimension of corresponding system of ODEs is. Furthermore, such problems are often stiff. This paper presents the algorithm of alternating order and step which is aimed at solving large-scale stiff problems with low accuracy. This algorithm has been included in the library of solvers [1] of the ISMA simulation environment.

Consider the Cauchy problem for the stiff system of ODEs

$$y' = f(t,y), \quad y(t_0) = y_0, \quad t_0 \le t \le t_k, \qquad (1)$$

where, y and f are sufficiently smooth real N-dimensional vector functions, t is an independent variable. Eigenvalues of its Jacobi matrix are pure real. It is well known that any initial value problem involving ODEs with higher derivatives can be reduced to this standard form. In [2-3] for the solution of (1) the explicit Runge-Kutta methods

$$y_{n+1} = y_n + \sum_{i=1}^{m} p_{mi} k_i, \quad k_i = hf\left(t_n + \alpha_i h, y_n + \sum_{j=1}^{i-1} \beta_{ij} k_j\right), \qquad (2)$$

are presented, where k_i, $1 \le i \le m$, are stages of the method, α_i, p_{mi}, β_{ij}, $1 \le i \le m$, $1 \le j \le i-1$, are numerical coefficients, defining accuracy and stability properties of scheme (2). Methods of form (2) are rather efficient on solving non-stiff problems. However, from the numerical results of solving stiff problems with integration algorithms based on explicit formulas which choose stepsize according to the required accuracy it follows that on the settling region (where derivatives of a solution are low) there is plenty of declined solutions. This is a result of appearing instability of a numerical scheme.

Algorithms based on explicit methods with stability control of a numerical scheme can solve this problem. In this case previous errors are suppressed due to stability control, whereas new errors are low due to low values of solution derivatives. As a result, the practical accuracy is even greater than the accuracy, which is required. Further improvement of the efficiency can be reached on application of methods with conformed stability domains.

In [3] the algorithm for obtaining coefficients of stability polynomials is presented. The use of these coefficients allows to design explicit Runge-Kutta m-stage methods for m equal up to 13 with defined form and size of a stability domain. It is also shown there that combining numerical formulas with different stability properties gives significant growth of performance. Transition from one numerical formula to another is performed according to stability criteria. At that, there is no explanation in [3] how to choose coefficients β_{ij} which affect stability of intermediate (inner) numerical schemes and, finally, the efficiency of the integration algorithm. The authors just noted that the stability of intermediate formulas can be

achieved, if β_{ij} are chosen sufficiently small. Below the method for choice of coefficients β_{ij} is offered.

II. Numerical Schemes

For simplicity, here is considered the Cauchy problem for autonomous system of ODEs

$$y' = f(y), \quad y(t_0) = y_0, \quad t_0 \le t \le t_k, \tag{3}$$

but all the findings that are to obtained below stay true for non-autonomous problems, if the coefficients in (2) are defined by the formulas

$$\alpha_i = \sum_{j=1}^{i-1} \beta_{ij}, \quad 2 \le i \le m, \quad \alpha_1 = 0. \tag{4}$$

To solve problem (3) the Runge-Kutta methods of the following form

$$\begin{aligned} y_{n,i} &= y_n + \sum_{j=1}^{i} \beta_{i+1,j} k_j, 1 \le i \le m-1, \\ y_{n+1} &= y_n + \sum_{i=1}^{m} p_{mi} k_i, \end{aligned} \tag{5}$$

can be applied, where $k_i = hf(y_{n,i-1})$, $1 \le i \le m$, $y_{n,0} = y_n$, and $y_{n,i}$ are defined by formulas (2).

Introduce matrix B_m with elements b_{ij} [3]

$$\begin{aligned} &b_{1i} = 1, \ 1 \le i \le m, \quad b_{ki} = 0, \ 2 \le k \le m, \ 1 \le i \le k-1, \\ &b_{ki} = \sum_{j=k-1}^{i-1} \beta_{ij} b_{k-1,j}, \ 2 \le k \le m, \ k \le i \le m, \end{aligned} \tag{6}$$

where β_{ij} are coefficients of scheme (2) or (5). It is to be used in the remainder of this paper.

Study stability on the linear scalar Dahlquist equation

$$y' = \lambda y, \quad y(0) = y_0, \quad t \ge 0, \tag{7}$$

with complex λ, $Re(\lambda) < 0$ (see [4]). Applying the second formula of (5) to (7), get

$$\begin{aligned} &y_{n+1} = Q_m(z) y_n, \ z = h\lambda, \ Q_m(z) = 1 + \sum_{i=1}^{m} c_{mi} z^i, \\ &c_{mi} = \sum_{j=i}^{m} b_{ij} p_{mj}, \ 1 \le i \le m \end{aligned} \tag{8}$$

In the notations $C_m = (c_{m1}, \ldots, c_{mm})^T$ and $P_m = (p_{m1}, \ldots, p_{mm})^T$, the third relation of (8) can be written in the form

$$B_m P_m = C_m, \tag{9}$$

where the elements of matrix B_m are defined by relations (6). For intermediate numerical schemes (4) we have

$$\begin{aligned} &y_{n,k} = Q_k(z) y_n, \ Q_k(z) = 1 + \sum_{i=1}^{k} c_{ki} z^i, \\ &c_{ki} = \sum_{j=i}^{k} b_{ij} \beta_{k+1,j}, \ 1 \le k \le m-1. \end{aligned} \tag{10}$$

On $\beta_k = (\beta_{k+1,1}, \ldots, \beta_{k+1,k})^T$ and $c_k = (c_{k1}, \ldots, c_{kk})^T$ coefficients β_{ij} of numerical schemes (5) and the coefficients in the corresponding stability polynomials satisfy the equation

$$B_k \beta_k = c_k, \quad 1 \le k \le m-1. \tag{11}$$

From the comparison between (6) and (10) it follows that $b_{ki} = c_{i-1,k-1}$, i.e. the elements of $(k+1)$-th column of matrix B_m equal to coefficients of stability polynomial $Q_k(z)$. Hence, if the coefficients of stability polynomials of basic and intermediate numerical schemes are defined, then the coefficients of methods (5) are unambiguously determined from linear systems (9) and (11) with upper triangular matrices B_i, $1 \le i \le m$.

Expansions of the exact and approximate solutions in the Taylor series in powers of h have the form

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + hf + 0.5h^2 f'f + O(h^3), \\ y_{n+1} &= y_n + \left(\sum_{j=1}^{m} b_{1j} p_{mj}\right) hf + \\ &\left(\sum_{j=2}^{m} b_{2j} p_{mj}\right) h^2 f'_n f_n + O(h^3), \end{aligned} \tag{12}$$

where the elementary differentials are computed on exact $y(t_n)$ and approximate y_n solutions, respectively. Comparison between relations (12) under assumption that $y(t_n) = y_n$, shows that numerical formula (5) has the first order of accuracy, if $\sum_{j=1}^{m} b_{1j} p_{mj} = 1$. Hence, to design m-stage methods of the first accuracy order, it is necessary to set $c_{m1} = 1$ in linear system (9).

III. Conformation of Stability Domains

Assume that the coefficients of the stability polynomials

$$Q_k(z) = 1 + \sum_{i=1}^{k} c_{ki} z^i, \quad 1 \le k \le m \tag{13}$$

are defined. Using approach from [5], we choose coefficients of the polynomial so that the stability domain expands along the imaginary axis and becomes singly connected. It provides better stability properties to rounding errors whereas the stability interval length reduces insignificantly.

For each k, $1 \le k \le m$, γ_k represents the length of such a maximal interval $[\gamma_k, 0]$, that for any $z \in [\gamma_k, 0]$ inequality $|Q_k(z)| \le 1$ satisfies. Taking into account, that $z = h\lambda$, in (13) for all $Q_k(z)$, $1 \le k \le m$ we replace h with $(h\gamma_k / \gamma_m)$. As a result, formula (13) may be written as follows

$$Q_k(z)=1+\sum_{i=1}^{k} c'_{ki} z^i,$$
$$c'_{ki}=(\gamma_k/\gamma_m)^i c_{ki},\quad 1\le k\le m. \tag{14}$$

The replacement of h with $(h\gamma_k / \gamma_m)$ means that the approximate solution obtained by intermediate schemes (5) is computed at points $(t_n + c'_{k1}h)$ $1 \le k \le m -1$ instead of $(t_n + c_{k1}h)$ $1 \le k \le m - 1$. In this case the maximal stepsize, obtained according to the stability requirements of the basic scheme is also maximal for intermediate numerical formulas.

Determine coefficients of methods (5) as follows. First, using [2] we compute coefficients of polynomials (13), satisfying some defined properties. Further, compute coefficients of polynomials (14) applying corresponding substitution of variables. Taking into account, that elements of $(k + 1)$-th column of matrix B_m coincide with coefficients of the stability polynomials $Q'_k(z)$, form matrix

$$B_m=\begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & c'_{11} & c'_{21} & \dots & c'_{m-1,1} \\ 0 & 0 & c'_{22} & \dots & c'_{m-1,2} \\ & & \dots & & \\ 0 & 0 & 0 & \dots & c'_{m-1,m-1} \end{pmatrix}, \tag{15}$$

Using in (11) vector $c'_k = (c'_{k1}, \dots, c'_{kk})^{\mathrm{T}}$ instead of c_k, we unambiguously determine all coefficients of methods (5) with conformed stability domains from linear system (9) and (11).

IV. Accuracy and Stability Control

We use the estimation of local truncation error $\delta_{n,1}$ to control accuracy of the first order methods. Applying (12) we get that for the m-stage method it has the form

$$\delta_{n,1}=(0.5-\sum_{j=2}^{m} b_{2j}p_{mj})h^2 f f' + O(h^3) = (0.5-c_{m2})h^2 f f' + O(h^3),$$

where c_{m2} is the coefficient at z^2 in stability polynomial (8). Estimation $\varepsilon_{n,1}$ of the error can be computed using the formula

$$\varepsilon_{n,1}=[(0.5-c_{m2})/(\alpha_i-\alpha_j)](k_i-k_j),\quad 1\le i,j\le m,\ i\ne j. \tag{16}$$

The graph of a solution of a stiff problem can be divided into two types of regions. The first one is the settling region (where values of solution derivatives are low), and the second one is the transition region (where values of solution derivatives are high). Taking this into account, to increase the performance of calculations we proceed as follows. We apply

$$\varepsilon'_{n,1}=[(0.5-c_{m2})/\alpha_2](k_2-k_1). \tag{17}$$

to make an over-cautious estimation. As k_1 linearly depends on integration stepsize, omission of inequality of $\|\varepsilon'_{n,1}\| \le \varepsilon$ leads just to one additional computation of the right part of (3). Here, ε is the absolute or relative tolerance of calculations, $\|\cdot\|$ is some norm in R^N. Taking into account, that

$$hf(y_{n+1})-k_1=h^2 f'_n f_n + O(h^3),$$

the final decision on accuracy we make checking inequality $\|\varepsilon''_{n,1}\| \le \varepsilon$, where

$$\varepsilon''_{n,1}=(0.5-c_{m2})(hf(y_{n+1})-k_1). \tag{18}$$

We construct the inequality for stability control similarly to [2]. To obtain this inequality we apply method (5) to problem (3) on $f(y) = Ay + b$, where A and b are N-dimensional matrix and vector with constant elements, respectively. As the result, we can estimate maximal eigenvalue λ_n^{max} of Jacobi matrix $\partial f(y_n)/ \partial y$ of (3) using the formula

$$h\lambda_n^{\max}=|\alpha_2\beta_{32}|^{-1}\max_{1\le j\le N}\left|\frac{[\alpha_2 k_3+\alpha_3 k_2-(\alpha_2+\alpha_3)k_1]_j}{[k_2-k_1]_j}\right|. \tag{19}$$

Then inequality for stability control for m-stage method (5) has form $h\lambda_n^{max} \le |\gamma_m|$, where $|\gamma_m|$ is stability interval length of the m-stage scheme.

V. First Order Method

For numerical solution of Cauchy problem (1) we consider the explicit five-stage Runge-Kutta method

$$\begin{gathered} y_{n+1}=y_n+p_1k_1+p_2k_2+p_3k_3+p_4k_4+p_5k_5,\\ k_1=hf(y_n),k_2=hf(y_n+\beta_{21}k_1),\\ k_3=hf(y_n+\beta_{31}k_1+\beta_{32}k_2),\\ k_4=hf(y_n+\beta_{41}k_1+\beta_{42}k_2+\beta_{43}k_3),\\ k_5=hf(y_n+\beta_{51}k_1+\beta_{52}k_2+\beta_{53}k_3+\beta_{54}k_4), \end{gathered} \tag{20}$$

where y and f are real N-dimensional vector functions, t is an independent variable, h is the integration step, k_1, k_2, k_3, k_4, and k_5 are stages of the method, p_1, p_2, p_3, p_4, p_5, β_{21}, β_{31}, β_{32}, β_{41}, β_{42}, β_{43}, β_{51}, β_{52}, β_{53}, β_{54} are numerical coefficients, defining accuracy and stability properties of (20).

We choose coefficients of (20) so that it has the first accuracy order and the extended stability domain. The stability domain of a method with the maximal length of the stability interval is almost multiconnected. We design polynomials of the first, second, third, fourth, and fifth degree so that the corresponding them methods have singly connected stability domains with the stability interval close to the maximal possible one (see fig. 1).

Applying the algorithm from [5], we get coefficients

$$c_{11}=c_{21}=c_{31}=c_{41}=c_{51}=1,$$

$$c_{22} = 0.128025128205128,$$

$$c_{32} = 0.152092927269786, c_{33} = 0.00580524400854353,$$

$$c_{42} = 0.160464544241005, c_{43} = 0.00827164513740441,$$
$$c_{44} = 0.000133419220894335,$$

$$c_{52} = 0.164341322127141, c_{53} = 0.00948975952580473,$$
$$c_{54} = 0.000223956930863224, c_{55} = 1.85097275222353 \cdot 10^{-6}.$$

At that,

$$\gamma_1 = -2, \quad \gamma_2 = -7.79, \quad \gamma_3 = -17.46,$$
$$\gamma_4 = -30.99, \gamma_5 = -48.39.$$

Writing and resolving linear systems (9) and (11) using (15), we obtain the coefficients of method (20)

$$\beta_{21} = 0.0413243016210550, \beta_{31} = 0.0805823881610573,$$
$$\beta_{32} = 0.0805823881610573, \beta_{41} = 0.1191668151228434,$$
$$\beta_{42} = 0.1597820013984078, \beta_{43} = 0.0819394878966193,$$
$$\beta_{51} = 0.1570787892802991, \beta_{52} = 0.2379583021959820,$$
$$\beta_{53} = 0.1631711307360486, \beta_{54} = 0.0822916178203657,$$

$$p_1 = 0.1945277188657676, p_2 = 0.3151822878089125,$$
$$p_3 = 0.2437005934695969, p_4 = 0.1641555613805598,$$
$$p_5 = 0.0824338384751631.$$

To control accuracy of the numerical formula we use estimations (17) and (18). The stability interval length of numerical scheme (20) of the first accuracy order equals 17.46. Therefore, for its stability control we can apply inequality $h\lambda_n^{max} \le 17.46$, where $h\lambda_n^{max}$ is defined by formula (19).

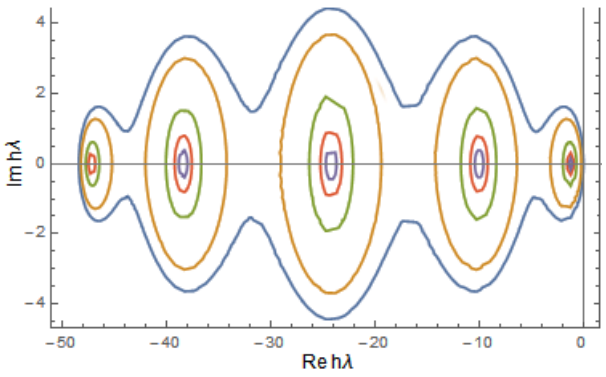

Figure 1. Stability domain of method (20).

VI. MERSON METHOD

The fourth accuracy order Merson method [6]

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{6}k_1 + \frac{2}{3}k_4 + \frac{1}{6}k_5, \\ k_1 &= hf(y_n), k_2 = hf(y_n + \frac{1}{3}k_1), \\ k_3 &= hf(y_n + \frac{1}{6}k_1 + \frac{1}{6}k_2), \\ k_4 &= hf(y_n + \frac{1}{8}k_1 + \frac{3}{8}k_3), \\ k_5 &= hf(y_n + \frac{1}{2}k_1 - \frac{3}{2}k_3 + 2k_4), \end{aligned} \tag{21}$$

is one of the most efficient and widely used explicit Runge-Kutta methods. The fifth computation of function f does not result in the fifth order of accuracy, but allows to extend the stability interval length to 3.5 and estimate truncation error $\delta_{n,4}$ using stages k_i, i.e.

$$\delta_{n,4} = (2k_1 - 9k_3 + 8k_4 - 2k_5)/30.$$

We apply inequality $\|\delta_{n,4}\| \le 5\varepsilon^{5/4}$ for accuracy control. The inequality is obtained assuming that the global error accumulated with local truncation errors [3]. Despite the fact that the inequality for accuracy control is obtained on a linear equation, it shows high reliability on solving non-linear problems.

Now let us construct the inequality for stability control. Applying to $k_3 - k_2$ the first order Taylor's formula with the remainder term written in the Lagrangian form, we have

$$k_3 - k_2 = h[\partial f(\mu_n)/\partial y](k_2 - k_1)/6,$$

where vector μ_n is computed in some vicinity of solution $y(t_n)$. Taking into account, that

$$k_2 - k_1 = h^2 f_n' f_n / 3 + O(h^3),$$

the inequality

$$v_{n,4} = 6 \cdot \max_{1 \le j \le N} \left| \frac{k_3^j - k_2^j}{k_2^j - k_1^j} \right| \le 3.5$$

can be used for stability control of (21), where 3.5 is the approximate length of stability interval (see fig. 2). Let $\varepsilon_{n,4} = \delta_{n,4}/5$. Then inequalities $\varepsilon_{n,4} \le 5\varepsilon^{5/4}$ and $v_{n,4} \le 3.5$ can be applied respectively for accuracy and stability control of scheme (21).

As estimation of eigenvalue $v_{n,4} = h\lambda_n^{max}$ is rough, stability control is used to limit integration stepsize and to

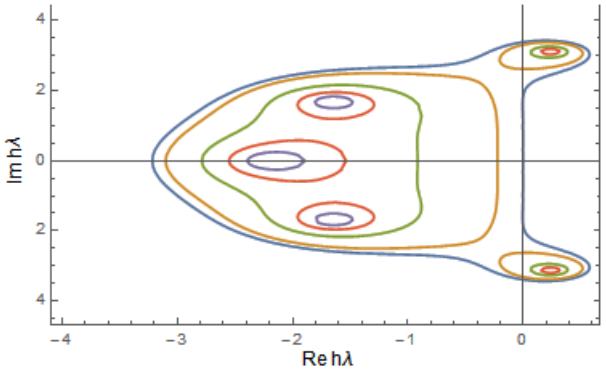

Figure 2. Stability domain of method (21).

switch between methods. The predicted step h_{n+1} is computed as follows. Step h^{ac}, that is chosen according to the requirements of accuracy, is computed using formula $h^{ac} = q_1 h_n$, where h_n is the latest accepted stepsize, and q_1, taking into account relation $\varepsilon_{n,4} = O(h_n^5)$, defined by $q_1^5 \varepsilon_{n,4} \le \varepsilon$. We compute step h^{st}, that is chosen according to the stability requirements, using $h^{st} = q_2 h_n$, where q_2, is defined by $q_2 \nu_{n,4} = 3.5$ as $\nu_{n,4} = O(h_n)$. Then, the predicted step h_{n+1} is computed using the formula

$$h_{n+1} = \max[h_n, \min(h^{ac}, h^{st})].$$

The given formula stabilizes stepsize over the settling region, where stability has the defining role.

VII. Integration Algorithm

The algorithm of alternating order and step can be easily formulated on a base of the developed methods. It chooses the most efficient scheme on an each step. Calculations are always begun with the Merson method as it is more accurate. Switch to the first order method with conformed stability domains is performed on omission of $\nu_{n,4} \le 3.5$. Transition to the Merson method is performed, if $\nu_{n,1} \le 3.5$ satisfies.

The norm in inequality for accuracy control is computed using the formula

$$\|\xi\| = \max_{1 \le i \le N} \frac{|\xi^i|}{|y_n^i| + r},$$

where i is a component number, r is a positive parameter. If inequality $\|y_n^i\| \le r$ satisfies for i-th component of a solution, absolute tolerance $r\varepsilon$ is controlled, otherwise, relative tolerance ε. On calculations r was assumed to be equal to 3.

VIII. Medical Akzo Nobel Problem

We chose the Medical Akzo Nobel problem [7] to test our method. The Akzo Nobel research laboratories formulated this problem in their study of the penetration of radio-labeled antibodies into a tissue that has been infected by a tumor. This study was carried out for diagnostic as well as therapeutic purposes.

In [7] there is considered a reaction diffusion system in one spatial dimension:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} - kuv, \quad \frac{\partial v}{\partial t} = -kuv, \tag{22}$$

which originates from the chemical reaction $A + B \to C$. Here, A, the radio-labeled antibody, reacts with substrate B, the tissue with the tumor, and k denotes the rate constant. The concentrations of A and B are denoted by u and v, respectively.

Making necessary transformations and defining $y(t)$ by $y = (u_1, v_1, u_2, v_2, \ldots, u_N, v_N)^T$ it is possible to write (22) in the form

$$\frac{dy}{dt} = f(t, y), \quad y(0) = g, \quad y \in R^{2N}, \quad 0 \le t \le 20, \tag{23}$$

Here, the integer N is a user-supplied parameter. The function f is given by

$$f_{2j-1} = \alpha_j \frac{y_{2j+1} - y_{2j-3}}{2\Delta\zeta} + \beta_j \frac{y_{2j-3} - 2y_{2j-1} + y_{2j+1}}{(\Delta\zeta)^2} - k y_{2j-1} y_{2j},$$
$$f_{2j} = -k y_{2j} y_{2j-1},$$

where

$$\alpha_j = 2(j\Delta\zeta - 1)^3 / c^2, \quad \beta_j = (j\Delta\zeta - 1)^4 / c^2, \quad 1 \le j \le N,$$
$$\Delta\zeta = 1/N, \quad y_{-1}(t) = \varphi(t), \quad y_{2,N+1} = y_{2,N-1}, \quad g \in R^{2N},$$
$$g = (0, v_0, 0, v_0, \ldots, 0, v_0)^T.$$

The function $\varphi(t) = 2$ at $0 < t \le 5$ and $\varphi(t) = 0$ at $5 < t \le 20$. Values for the parameters k, v_0, and c are 100, 1, and 4, respectively. Graph of the time and space dependencies of u and v is shown in fig. 3.

IX. Numerical Results

Calculations were performed on Intel(R) Core(TM) i3-5010U CPU with double precision. The parameter N was equal 200 that means that the system to be solved involved 400 equations.

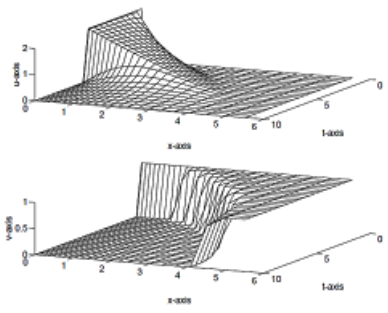

Figure 3. u and v as functions of time and space.

The stiffness ratio of the Medical Akzo Nobel problem approximately equals 10^6. The graph of 133rd component of the solution is shown in fig. 4.

Below *IS*, *IW*, and *IF* represent, respectively, total numbers of steps, declined solutions (due to omission of the defined absolute tolerance), and computed right parts of the problem.

The algorithm of alternating order and step based on the first order method with conformed stability domains and the Merson method with accuracy and stability control gives the following results. For the defined absolute tolerance equal to 10^{-4} we have *IS* = 11 505, *IW* = 1 266, and *IF* = 70 893. For the absolute tolerance 10^{-7}: *IS* = 72 658, *IW* = 10 333, and *IF* = 403 066.

X. CONCLUSION

From the numerical results it follows that stability control leads to the efficiency gain due to the reduction of some declined solutions appearing as a result of instability of a numerical formula. Simulation of other test examples gives similar tendency. The designed method is aimed at the solution of large-scale problems of moderate stiffness with low accuracy, as well as problems with protensive settling regions, where the first order methods with conformed stability domains give growth of the efficiency.

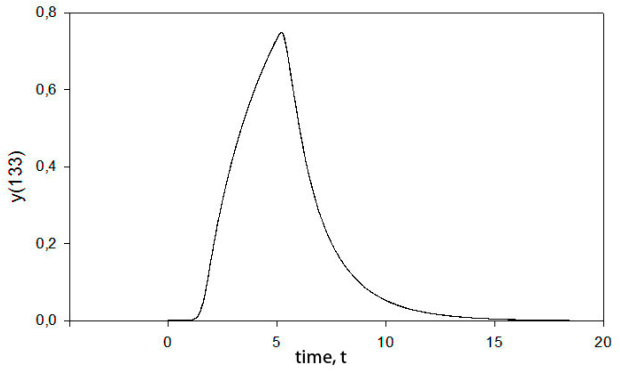

Figure 4. Solution of the Medical Akzo Nobel problem.

The constructed algorithm is designed for low precision calculations – about 1% and lower. In this case, its maximum efficiency is reached. In the algorithm, with its parameters, one can specify different modes of calculations: 1) with the explicit first order method with conformed stability domains either with or without stability control; 2) with the Merson method either with or without stability control; 3) with automatic choice of a numerical scheme. Therefore, this algorithm can be applied both for solving stiff and non-stiff problems. In calculations with automatic choice of a numerical scheme, the integration algorithm makes a decision whether a problem to be solved is stiff or not by itself.

ACKNOWLEDGMENT

This work is partially supported by Russian Foundation of Fundamental Researches (project 14-01-00047).

REFERENCES

[1] E.A. Novikov, Yu. V. Shornikov. Computer simulation of hybrid stiff systems. Novosibirsk: Publishing House of the Novosibirsk State Technical University, 2012. (in russian).

[2] E. Hairer, G. Wanner. Solving ordinary differential equations II. Stiff and differential-algebraic problems. Springer-Verlag, 1996.

[3] E.A. Novikov. Explicit methods for stiff systems. Novosibirsk: Nauka, 1997. (in russian).

[4] G. Dahlquist, "A special stability problem for linear multistep methods," BIT , vol. 3, 1963, pp. 23-43.

[5] E.A. Novikov, M.V. Rybkov. "Numerical algorithm for stability domains design for the first order methods," Vestnik Buryatskogo gosudarstvennogo universiteta, №9-2, 2014, pp. 80–85. (in Russian).

[6] R.H. Merson. "An operational methods for integration processes," Proc. Symp. Data Processing. Weapons Research Establishment, 1957, 331 p.

[7] F. Mazzia, C. Magherini. Test set for initial value problem solvers: monograph. Department of Mathematics, University of Bari and INdAM, Research Unit of Bari. Release 2.4, 2008.