

Numerical Algorithm for Design of Stability Polynomials for the First Order Methods

Eugeny A. Novikov¹ Mikhail V. Rybkov² Anton E. Novikov³

¹Institute of Computational Modelling, Federal Research Center, Russia, novikov@icm.krasn.ru

²Institute of Mathematics and Fundamental Informatics, Siberian Federal University, mixailrybkov@yandex.ru

³Institute of Mathematics and Fundamental Informatics, Siberian Federal University, Russia, aenovikov@bk.ru

Abstract

This paper derives an algorithm for computing coefficients for stability polynomials of a degree up to $m = 35$. These coefficients correspond to explicit first order Runge-Kutta methods. Authors showed dependence between stability polynomial values at extreme points and both size and form of a stability domain. Numerical results are given.

Keywords: stiff problem, explicit methods, stability polynomials

1 Introduction

Heterogeneous algorithms are applied to solving stiff problems in a number of situations. Such algorithms are designed using the fact that on the settling and transition regions integration stepsizes are limited according to the requirements of stability and accuracy, respectively. Efficiency growth is achieved by applying an explicit scheme over the transition region and an L-stable scheme over the settling region. The switch between methods is performed using an inequality for stability control.

The problem is that the size of stability domains of the known methods is too small. Some monographs and papers present explicit methods with extended stability domains (Novikov, Shornikov, 2012). The way to obtain stability polynomials providing the maximal length of a stability domain is considered in (Skvortzov, 2011). Novikov (1997) proposed an algorithm which provides polynomials coefficients. The algorithm allows to design explicit Runge-Kutta methods with specified stability domain forms and sizes. Furthermore, coefficients of stability polynomials having a degree up to $m = 13$ are found there.

Here we develop an algorithm that provides stability polynomials coefficients having a degree up to $m = 27$. The coefficients correspond to explicit first order Runge-Kutta methods. It is shown that the form, size, and structure of a stability domain depend on the position of the stability polynomial roots on the complex plane.

2 Explicit Runge-Kutta Methods

To solve the stiff problem

$$y' = f(t, y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq t_k,$$

where y and f are smooth real N -dimensional vector-functions, t is an independent variable, in (Novikov, 1997) explicit methods

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

are considered, where k_i , $1 \leq i \leq m$, are stages of the method, h is the integration stepsize, p_{mi} , α_{ij} , and β_{ij} are numerical coefficients defining accuracy and stability properties of this numerical scheme. For simplicity, let us consider the following Cauchy problem for the autonomous system of ODEs

$$y' = f(y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq t_k. \quad (1)$$

We apply methods of the form

$$y_{n,i} = y_n + \sum_{j=1}^i \beta_{i+1,j} k_j, \quad 1 \leq i \leq m-1, \\ y_{n+1} = y_n + \sum_{i=1}^m p_{mi} k_i, \quad (2)$$

to solve (1), where $k_i = hf(y_{n,i-1})$, $1 \leq i \leq m$, $y_{n,0} = y_n$. All the findings those are to obtained below can be used for non-autonomous problems, if

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

Stability of one-step methods is widely studied on the Dahlquist equation $y' = \lambda y$, $y(0) = y_0$, $t \geq 0$ with complex λ , $\text{Re}(\lambda) < 0$ (Dahlquist, 1963). Applying the second formula from (2) to solve $y' = \lambda y$, we get

$$y_{n+1} = Q_m(z) y_n, \quad Q_m(z) = 1 + \sum_{i=1}^m c_{mi} z^i, \\ c_{mi} = \sum_{j=1}^m b_{ij} p_{mj}, \quad 1 \leq i \leq m,$$

where $z = h\lambda$. Hence, the stability function of a m -stage explicit Runge-Kutta method is polynomial $Q_m(z)$ of a degree m . Novikov (1997) gave order conditions for methods of form (2) and, in particular, method (2)

has the first accuracy order, if $p_{m1} + \dots + p_{mm} = c_{m1} = 1$. Further, we consider the problem of finding such coefficients that a stability domain has specified form and size.

3 Stability Polynomials Over $[\gamma_m, 0]$

Let k and m be given integers, $k \leq m$. Consider polynomials

$$Q_{m,k}(x) = 1 + \sum_{i=1}^k c_i x^i + \sum_{i=k+1}^m c_i x^i, \quad (3)$$

where $c_i, 1 \leq i \leq k$, are defined, and $c_i, k+1 \leq i \leq m$, are arbitrary. Usually $c_i, 1 \leq i \leq k$, are determined according to the requirements of accuracy. Therefore, let us assume that $c_i = 1/i!, 1 \leq i \leq k$.

Denote extreme points of (3) by x_1, \dots, x_{m-1} , at that $x_1 > x_2 > \dots > x_{m-1}$. Define unknown coefficients $c_i, k+1 \leq i \leq m$, so that polynomial (3) has predefined values at extreme points $x_i, k \leq i \leq m-1$, i.e. $Q_{m,k}(x_i) = F_i, k \leq i \leq m-1$, where $F(x)$ is some given function, $F_i = F(x_i)$. For this purpose, consider the following system of algebraic equations

$$Q_{m,k}(x_i) = F_i, Q'_{m,k}(x_i) = 0, k \leq i \leq m-1, \quad (4)$$

$$Q'_{m,k} = \sum_{i=1}^m i c_i x^{i-1},$$

in variables $x_i, k \leq i \leq m-1$, and $c_j, k+1 \leq j \leq m$.

Rewrite (4) in the form, suitable for calculations on the computer. Denote through y, z, g , and r vectors with components

$$y_i = x_{k+i-1}, z_i = c_{k+i}, g_i = F_{k+i-1} - 1 - \sum_{j=1}^k c_j y_i^j,$$

$$r_i = -\sum_{j=1}^k j c_j y_i^{j-1}, 1 \leq i \leq m-k,$$

through E_1, E_2, E_3 – diagonal matrices with elements

$$e_1^{ii} = k+i, e_2^{ii} = 1/y_i,$$

$$e_3^{ii} = (-1)^{k+i-1}, 1 \leq i \leq m-k,$$

and through A – a matrix with elements $a_{ij} = y_i^{k+j}, 1 \leq i, j \leq m-k$. Using these notations problem (4) can be written as follows

$$Az - g = 0, E_2 A E_1 z - r = 0. \quad (5)$$

System (5) is ill-conditioned that leads to some difficulties while solving it with the fixed point iteration method. For convergence of the Newton's method, it is necessary to somehow obtain good initial values that in this case is a separate difficult problem. If we assume in (4) that $F_i = (-1)^i, k \leq i \leq m-1$, we find the polynomial with the maximal length of the stability interval. In this case the problem of computation of initial value y^0 is solved using values of the Chebyshev polynomial at extreme points over interval $[-2m^2, 0]$, where m is a degree of polynomial (3). Those values can be computed using the formula

$$y_i = m^2 [\cos(i\pi/m) - 1], 1 \leq i \leq m-1. \quad (6)$$

Substituting (6) in the system (5), get coefficients of the Chebyshev polynomial, for those $|Q_{m1}(x)| \leq 1$ on $x \in [-2m^2, 0]$. For any k we can take (6) as initial values

and, as numerical results show, there is good convergence rate in this case. If $F_i \neq (-1)^i, k \leq i \leq m-1$, then the choice of initial values is a separate difficult problem.

Let us describe a way to solve (5) that does not require good initial values. Apply the relaxations for the numerical solution of (5). The main idea of the relaxations is that for a steady-state problem we run unsteady-state process which solution settles to the solution of the initial problem. Consider the Cauchy problem

$$y' = E_3(E_2 A E_1 A^{-1} g - r), y(0) = y_0. \quad (7)$$

Apparently, after a stationary point of (7) has been found, the stability polynomial coefficients can be computed from the system (5). Notice, that due to using matrix E_3 all eigenvalues of the Jacobi matrix of (7) have negative real components, i.e. problem (7) is stable. Numerical results show that (7) is a stiff problem. Applying methods that require evaluation of the Jacobi matrix may cause difficulties while solving (7). Therefore, we solve (7) with the second accuracy order method that uses numerical computing and freezing the Jacobi matrix (Novikov, 2008). When applied to the problem $y' = f(y), y(0) = y_0$, this method takes the form

$$y_{n+1} = y_n + a k_1 + (1-a)k_2, D_n = E - a h_n A_n,$$

$$D_n k_1 = h_n f(y_n), D_n k_2 = k_1. \quad (8)$$

Here, $a = 1 - 0.5\sqrt{2}, k_1$ and k_2 are stages of the method, E is the identity matrix, h_n is the integration stepsize, A_n is a matrix representable in the form $A_n = f_n' + h_n R_n + O(h_n^2), f_n' = \partial f(y_n)/\partial y$ is the Jacobi matrix of (7), R_n is the integration stepsize independent matrix. Since matrix R_n is arbitrary, problems of numerical solving and freezing the Jacobi matrix can be concerned simultaneously. To control accuracy of (8) we apply the inequality

$$\varepsilon(j_n) = \|D_n^{-j_n}(k_2 - k_1)\| \leq a\varepsilon/|a-1/3|, 1 \leq j_n \leq 2, \quad (9)$$

where ε is the required accuracy of calculations, $\|\cdot\|$ is some norm in R^N , and integer variable j_n is chosen minimum for which inequality (9) is satisfied. The numerical differentiation step $s_j, 1 \leq j \leq N$, is chosen using the formula $s_j = \max\{10^{-14}, 10^{-7}|y_j|\}$. In this case j -th column a_n^j of matrix A_n is computed using the formula

$$a_n^j = [f(y_1, \dots, y_j + s_j, \dots, y_N) - f(y_1, \dots, y_j, \dots, y_N)]/s_j,$$

$$1 \leq j \leq N,$$

i.e. it is required to perform N computations of the right part of problem (7) to define A_n . An attempt to use previous matrix D_n is performed after each successful integration step. To preserve stability properties of the numerical scheme, on freezing matrix D_n the integration stepsize is kept permanent. Recomputation of the matrix is carried out in the following cases: 1) calculations accuracy is

degenerated, 2) quantity of steps with a frozen matrix has reached maximal number I_h , 3) the predicted step is greater than the previous successful one in Q_h times.

4 Stability Polynomials Over [-1, 1]

It is not difficult to see that stability polynomial coefficients approach zero as m increases. Novikov (1997) presented coefficients c_i , $k + 1 \leq i \leq m$, for polynomials of a degree up to $m = 13$. Now consider an algorithm providing polynomials with specified properties over the interval $[-1, 1]$. In this case coefficients c_i grow not that much, and it is possible to derive polynomials for $m > 13$. Denote through $|\gamma_m|$ the length of stability interval of m -stage explicit formula of the Runge-Kutta type, i.e. the inequality $|Q_{m,k}(x)| \leq 1$ over the interval $[\gamma_m, 0]$ is satisfied. Then, substituting $x = 1 - 2z/\gamma_m$ we can map $[\gamma_m, 0]$ into $[-1, 1]$ and obtain polynomial

$$Q_m(z) = \sum_{i=0}^m d_i z^i \tag{10}$$

Coefficients d_i , $0 \leq i \leq m$ of polynomial (10) and coefficients c_i , $0 \leq i \leq m$, of (3) satisfy the relation

$$c = UVd, \tag{11}$$

where $d = (d_0, \dots, d_m)^T$, $c = (c_0, \dots, c_m)^T$, U is a diagonal matrix with elements $u^{ii} = (-2/\gamma_m)^{i-1}$, $1 \leq i \leq m + 1$. Elements v^{ij} of V are defined by

$$v^{lj} = 1, 1 \leq j \leq m+1; v^{ij} = v^{i,j-1} + v^{i-1,j-1}, \\ 2 \leq i \leq j \leq m+1; v^{ij} = 0, i > j.$$

Obviously, V represents the Pascal's triangle which elements are easily computed using a recurrent formula. Therefore, after deriving the polynomial (10) over interval $[-1, 1]$, using (11) it is easy to compute coefficients of the polynomial (3).

Now let us derive polynomial (10). We denote the extreme points of (10) through z_1, \dots, z_{m-1} , at that $z_1 > z_2 > \dots > z_{m-1}$. We compute coefficients d_i , $0 \leq i \leq m$, under condition that polynomial (10) has predefined values in extreme points z_i , $1 \leq i \leq m - 1$, i.e.

$$Q_m(z_i) = F_i, 1 \leq i \leq m-1,$$

where $F(z)$ is some given function, $F_i = F(z_i)$. For that, consider the following system of algebraic equations

$$Q_m(z_i) = F_i, Q'_m(z_i) = 0, 1 \leq i \leq m-1, \\ Q'_m(z) = \sum_{i=1}^m i d_i z^{i-1}, \tag{12}$$

here the normality conditions $Q_m(-1) = (-1)^m$ and $Q_m(1) = 1$ are satisfied.

Rewrite (12) in the form, suitable for calculations on the computer. For this purpose, denote by y , w , g , and r vectors with components

$$y_j = z_j, r_j = 0, 1 \leq j \leq m-1; w_i = d_{i-1}, 1 \leq i \leq m+1, \\ g_i = F_i, 1 \leq i \leq m-1; g_i = 1, i = m; \\ g_i = (-1)^m, i = m+1;$$

through E_1 and E_2 matrices of dimension $(m + 1) \times (m + 1)$ and $(m - 1) \times (m + 1)$, respectively, with elements of the form

$$e_1^{jj} = j-1, 1 \leq j \leq m+1; e_2^{ii} = 1/y_i, 1 \leq i \leq m-1,$$

and through A – a matrix of dimension $(m + 1) \times (m + 1)$ with elements

$$a^{ij} = y_i^{j-1}, 1 \leq i \leq m-1, 1 \leq j \leq m+1; a^{m,j} = 1, \\ a^{m+1,j} = (-1)^{j+1}, 1 \leq j \leq m+1.$$

Now problem (12) can be written as follows

$$Aw - g = 0, E_2 A E_1 w - r = 0. \tag{13}$$

For the numerical solution of (13) we use the relaxations (Novikov, 1997). After the determination of polynomial (10) coefficients, compute the coefficients of polynomial (3) using relation (11). Find value γ_m under assumption that the polynomial to be obtained corresponds to the first order method, i.e. $c_1 = 1$. Having written the second relation and having made necessary transformations, we get

$$\gamma_m = \left\{ -2 \sum_{j=1}^{m+1} v_{2j} d_j \right\} / c_1, \gamma_m^0 = -2m^2.$$

5 Form and Size of Stability Domains

Let us describe how the choice of values F_i affects the size and form of a stability domain. If we let $F_i = (-1)^i$, $k \leq i \leq m - 1$, then the stability interval length is known and computed using the formula $|\gamma_m| = 2m^2$. In this case for given m we get the maximal length of a stability domain along the real axis. Figure 1 shows level curves $|Q_{m,k}(x)| = 1$, $|Q_{m,k}(x)| = 0.8$, $|Q_{m,k}(x)| = 0.6$, $|Q_{m,k}(x)| = 0.4$, and $|Q_{m,k}(x)| = 0.2$ on the complex plane $\{h\lambda\}$ for the stability domain on $m = 4$, $k = 1$, $F = \{-1, 1, -1\}$. The stability interval length $|\gamma_m|$ of the corresponding method equals 32.

In case if the stability interval length is maximal, a stability domain is almost multiconnected, so rounding errors may lead to stepping out of the stability domain. To solve this problem we need to stretch the stability domain along the imaginary axis in tangency points of parts of the stability domain. For this purpose, we can let $F_i = (-1)^i \mu$, $1 \leq i \leq m - 1$, $0 < \mu < 1$. Numerical results show that if $\mu = 0.9$, the stability interval length becomes shorter by 5–8% comparing to the maximal possible length equal to $2m^2$. At that, the stability domain stretches along imaginary axis at the tangency points. This provides better stability properties of the corresponding method to rounding errors on insignificant reduction of the stability interval length. If we assume $\mu = 0.95$, then the stability interval length is reduced by 3–4%. The stability domain of the five-stage method on $\mu = 0.9$ is shown in Figure 2. The stability interval length of this method $|\gamma_m| = 30.00$.

As μ decreases from 1 to 0, roots of polynomial (3) get closer to each other on the real axis. Therefore, the stability interval length is reduced. The ellipses,

which are well-defined on $\mu = 1$ get closer not providing essential stretch of the stability domain along the imaginary axis. Therefore, depending on the problem to be solved it is reasonable to choose value μ from 0.8 to 0.95.

On solving problems, which Jacobi matrices have eigenvalues with imaginary components and which solutions have an oscillating behavior, the extension of a stability interval is not always obligatory. In this case, the integration stepsize is rather small due to the accuracy requirements and thus it is more reasonable to extend a stability domain along the imaginary axis. If the Jacobi matrix have pure imaginary eigenvalues, it is necessary to have the condition $|Q_{m,k}(x)| = 1$ satisfied over some region on the imaginary axis. This requirement is satisfied as k increases.

For the first order methods, i.e. for $k = 1$, it is possible to make the requirement satisfied choosing appropriate values of function F . For instance, on $m = 4, k = 1, F = \{0.75, 0.80, 0.75\}$ we obtain a polynomial, satisfying this requirement (Figure 3). Since m is even and all the values F_i are positive, the graph of the polynomial does not cross the real axis, at that, polynomial has two pairs of complex conjugate roots. Therefore, the stability domain stretches along the imaginary axis and some region of the imaginary axis belongs to the stability domain. The stability interval length equals 2.89. On reducing values F_i the length of a stability domain along the real axis gets greater. While further reduction of values F_i the stability interval length $|\gamma_m|$ also becomes greater but the region on the imaginary axis belonging to the stability domain becomes less. Therefore, for developing first order methods aimed at solving oscillating problems, it is reasonable to choose stability polynomials which have a couple of complex conjugate roots in a complex plane $\{h\lambda\}$ nearby the origin of coordinates. At that, values F_i that correspond to these roots are needed to be chosen close to 1, so that the stability domain has the maximal region of the imaginary axis in it.

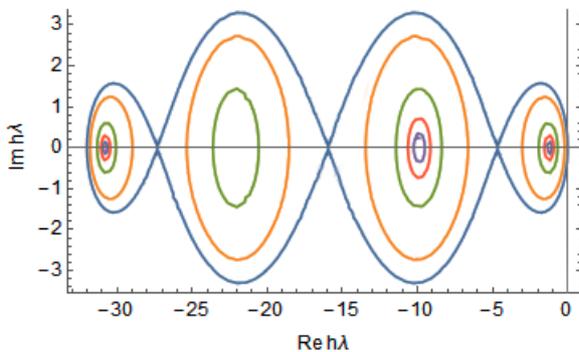


Figure 1. Stability domain on parameters $m = 4, k = 1, F = \{-1, 1, -1\}$.

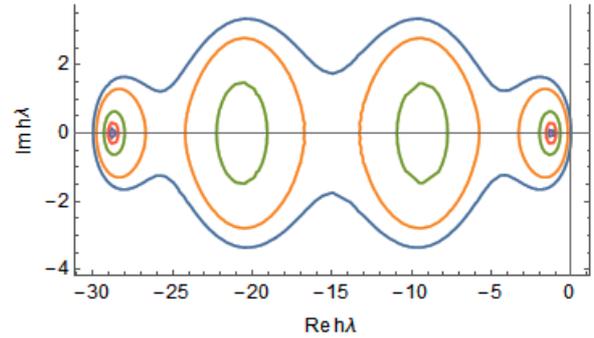


Figure 2. Stability domain on parameters $m = 4, k = 1, F = \{-0.9, 0.9, -0.9\}$.

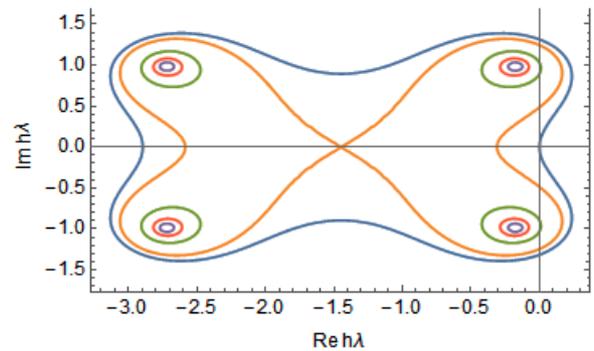


Figure 3. Stability domain on parameters $m = 4, k = 1, F = \{0.75, 0.80, 0.75\}$.

6 Numerical Results

The numerical results show that coefficient c_m of polynomial (3) reduces as m grows. In particular, on $m = 13$ and $k = 1$ value c_m is of the order of 10^{-26} . It is difficult to solve problem (7) for $m > 13$ due to rounding errors. Numerical results of solving (11) show that polynomial (10) coefficients $d_i, 0 \leq i \leq m$, grow in magnitude simultaneously with m . In particular, on $m = 13$ value $\max_{0 \leq i \leq m} |d_i|$ is of the order of 10^5 , and on $m = 25$ of the order of 10^9 , i.e. d_i grow slower than c_i . We transit from polynomial (10) coefficients to coefficients of (3) using (11) after (13) has been solved. This allows to compute the coefficients of stability polynomials of a degree up to $m = 27$.

It is difficult to solve problem (11) with double precision for $m > 27$ due to the appearing rounding errors. The algorithm using the "Quade-Double Precision Library" (described in (Hida, 2000)) was developed to compute the stability polynomials coefficients with greater m .

The "QD Precision Library" allows performing calculations with higher accuracy. While the standard data type 'double', allowing to represent numbers with double precision, is confined to 53 bits of the binary mantissa and provides precision about 16 decimal numerals, numbers of the data type 'dd_real' from the

library QD has the 106-bit mantissa that provides precision about 32 decimal numerals. In fact, the number of the type `dd_real` is the software-implemented sum of two numbers of the type 'double'. At that, the mantissa of the sum elongates in two times, but the range of values, presentable in new data type does not change and the possible values vary from about 10^{-308} to 10^{308} , as for the standard 'double'. Despite the confinement, accuracy of the representation of numbers in this diapason increases.

On the implementation of the algorithm for computation of the coefficients of (8) using the data type 'dd_real' the main input parameters of the algorithm – accuracy of calculations ε and differentiation stepsize s_j did not change. The Chebyshev polynomial values at the extreme points were chosen for initial conditions. The improved precision of the numbers representation allowed to compute polynomial coefficients for $m > 27$.

7 Conclusions

Authors of this article computed the coefficients for stability polynomials of a degree up to $m = 35$ using an algorithm providing polynomials over the interval $[-1, 1]$. These coefficients correspond to the first order Runge-Kutta methods with specified both form and size of stability domains. It is shown that the choice of values of stability polynomials at extreme points affects form and size of a stability domain. The proposed algorithm for designing stability domains increases the efficiency of explicit methods. Furthermore, it allows to develop algorithms of alternating order and step for solving problems of moderate stiffness. If the solution behavior of a problem which is to be solved is known, then it is possible to design an integration algorithm with the stability domain suitable for the given class of problems.

From our point of view, one of the main future applications of these results is to use the proposed algorithm to design numerical methods for solving ODEs systems. These methods can be included in libraries for software aimed at computer simulation.

Acknowledgements

This work is partially supported by Russian Foundation of Fundamental Researches (project №17-07-01513A).

References

- G. Dahlquist: A Special Stability Problem for Linear Multistep Methods. *BIT*, 3: 23–43, 1963.
- Y. Hida, X. S. Li, and D. H. Bailey. *Quad-Double Arithmetic: Algorithms, Implementation, and Application*. Technical Report LBNL-46996, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, 2000.

- E. A. Novikov. *Explicit Methods for Stiff Systems*. Novosibirsk: Nauka, 1997. (in Russian)
- E. A. Novikov and Y. V. Shornikov. *Computer Simulation of Hybrid Stiff Systems*. Novosibirsk: Publishing House of the Novosibirsk State Technical University, 2012. (in Russian)
- A. E. Novikov and E. A. Novikov: L-stable (2,1)-method for solving stiff non-autonomous problems. *Computational Technologies*, 13: 477–482, 2008. (in Russian)
- L. M. Skvortzov: Simple way to Design Stability Polynomials for Explicit Stabilized Runge-Kutta Methods. *Matematicheskoe modelirovanie*, 23(1): 81–86, 2011.