Interval Computations
No 3(5), 1992

TWO-SIDED METHOD FOR SOLVING SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS WITH AUTOMATIC DETERMINATION OF GUARANTEED ESTIMATES

Oleg B. Ermakov

A way of constructing two-sided approximations to a solution of a system of ODE based on the Adams extrapolation method of any order with automatic facilities for taking into account errors of input data, errors of procedure, and rounding errors using the PASCAL-XSC precompiler is considered.

ДВУСТОРОННИЙ МЕТОД РЕШЕНИЯ СИСТЕМЫ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ С АВТОМАТИЗИРОВАННЫМ ПОЛУЧЕНИЕМ ГАРАНТИРОВАННЫХ ОЦЕНОК

О.Б.Ермаков

Рассматривается способ построения двусторонних приближений решения системы ОДУ на основе экстраполяционного метода Адамса любого порядка с возможностью автоматизированного учета погрешностей входных данных, погрешности метода и погрешности округлений с использованием прекомпилятора PASCAL-XSC.

The present approach is based theoretically on the work [1] that states that to obtain lower and upper estimates for a scalar differential equa-

tion, in a computational formula of the Adams method, the precomputed constant is added.

This constant must compensate the error of procedure and the rounding error: with excess for the upper function and with defect for the lower function. A generalization of [1] for the case of an ODE system was considered in [4].

In the present paper, one possible way of actual construction of upper and lower functions for an ODE system with ill-posed initial data using the PASCAL-XSC compiler is described.

For the given ODE system

$$y'(t) = f(t, y), \tag{1}$$

$$y(t_0) \in [y_0], \tag{2}$$

where $y, f \in R^n$, $[y_0] := [\underline{y}_0, \overline{y}_0] \in I(R^n)$, is supposed that the right-hand sides of (1) satisfy the condition of nondiagonal monotonicity $\partial f_i / \partial y_j \geqslant 0, i \neq j$. For example, the ODE of such form arise when describing chemical kinetics reactions. In the same way [4], to obtain a guaranteed two-sided estimate for a solution of system (1), the ODE system of the following form is introduced:

$$z'(t) = f(t, z) + q, \tag{3}$$

$$z(t_0) \in [y_0] \subseteq [z_0], \tag{4}$$

where $q \in I(R^n)$. The desired interval vector q must be chosen in such a manner that the following inclusions hold:

$$y(t_k) \in [z(t_k)]. \tag{5}$$

Let $\widetilde{y}_k \in R^n$ be a vector of approximated values for $y_k := y(t_k)$. $k = \overline{0, m-1}$, and for $k = \overline{m, p}$, we have

$$\widetilde{y}_k = \widetilde{y}_{k-1} \oplus h \otimes \sum_{\nu=1}^{m} c_\nu \otimes f(t_{k-\nu}, \widetilde{y}_{k-\nu}), \tag{6}$$

where $\oplus, \otimes$ are machine arithmetic operations. For approximate values of lower and upper functions of system (3), we have for $k = \overline{m, p}$

$$[\widetilde{z}_k] = [\widetilde{z}_{k-1}] \oplus h \otimes \sum_{\nu=1}^{m} c_\nu \otimes (F(t_{k-\nu}, [\widetilde{z}_{k-\nu}]) \dotplus q), \tag{7}$$

where $F(t_{k-\nu}, [\widetilde{z}_{k-\nu}])$ — natural interval extension of f.

Denoting by $\widetilde{y}_{k-1}(t) \in R^n$ an exact solution of system (1) with the initial condition

$$\widetilde{y}_{k-1}(t_{k-1}) = \begin{cases} \overline{\widetilde{z}}_{k-1}, & \text{when constructing upper functions} \\ \underline{\widetilde{z}}_{k-1}, & \text{when constructing lower functions} \end{cases}$$

and taking into account the actual error $\Delta_k \in R^n$ ([4]) in computing values of upper and lower functions respectively, the following statement, from which the correction vector $q \in I(R^n)$ of the right-hand side of system (3) is obtained.

Theorem. *In the domain $D \subseteq R^{n+1}$, let the following conditions hold:*

1) $f_i(t, y_1, \ldots, y_n) \in C_D^m, i = \overline{1,n}$;

2) *the values $\overline{\widetilde{z}}_{i,k}, \underline{\widetilde{z}}_{i,k}$ computed according to (6) are such that the points $(t_k, \overline{\widetilde{z}}_{i,k}), (t_k, \underline{\widetilde{z}}_{i,k}) \in D, k = \overline{m,p}$, moreover, $\forall k = \overline{0, p-1}, t_k \leqslant t \leqslant t_{k+1}$, $\widetilde{y}_{i,k}(t) \leqslant \widetilde{y}_i \leqslant \widetilde{y}_{i,k+m-1}(t)$ is contained in D;*

3) $\beta_{i,k} + \rho_{i,k} \in [\underline{\gamma}_i, \overline{\gamma}_i] \subseteq [\underline{\gamma}, \overline{\gamma}], k = \overline{m,p}$, $\partial f_i/\partial y_j \in [\underline{l}_i, \overline{l}_i] \subseteq [\underline{l}, \overline{l}]$,

$$\left| \partial^2 f_i/\partial t \partial y_j + \sum_{\sigma=1}^{n} (\partial^2 f_i/\partial y_j \partial y_\sigma) f_\sigma + \sum_{\tau=1}^{n} (\partial f_i/\partial y_\tau)(\partial f_\tau/\partial y_j) \right| \leqslant M;$$

4) *the initial values $\overline{\widetilde{z}}_{i,k}, \underline{\widetilde{z}}_{i,k}, k = \overline{1, m-1}$, satisfy the inclusions $[y_0] \subseteq [z_0]$,*

$$\overline{\widetilde{z}}_{i,k} - \widetilde{y}_{i,k-1}(t_k) \in [0, \Delta_i] \subseteq [0, \Delta],$$
$$\underline{\widetilde{z}}_{i,k} - \widetilde{y}_{i,k-1}(t_k) \in [-\Delta_i, 0] \subseteq [-\Delta, 0];$$

5) *the quantities Δ, $h\overline{q}_i, h\underline{q}_i$ are such that*

$$\Delta \geqslant (\overline{\gamma} - \underline{\gamma})/(1 - nh(\overline{l} - \underline{l})(\eta^+ - \eta^-)\exp(-nh\underline{l}) - 2nh^2 b),$$
$$h\overline{q}_i \geqslant -(\overline{\gamma}_i + \underline{\gamma}_i)/2 + (\Delta/2)(1 - nh((\overline{l}_i + \underline{l}_i)/2)\exp(-nh\underline{l})),$$
$$h\underline{q}_i \leqslant (\overline{\gamma}_i + \underline{\gamma}_i)/2 - (\Delta/2)(1 - nh((\overline{l}_i + \underline{l}_i)/2)\exp(-nh\underline{l})),$$

where η^+, η^-, b are some quantities.

Then at points $t_k \in [t]$, $k = \overline{1,p}$, *the solution for the problem (1)–(2) is*

$$y(t_k) \in [\widetilde{z}_k].$$

This theorem is a generalization of [4]; it takes into account aside from rounding errors and the Adams method errors of procedure, input data errors.

We dwell on some details of the practical realization of the above approach with the use of the facilities of PASCAL-XSC compiler. We note that in the solving an ODE system with initial conditions, by Adams methods, formulas of the form (6) are usually used which are represented by means of the functions of the right-hand side of the system and are obtained using finite differences. Therefore, to increase the accuracy of the result and obtain a simple realization, it is appropriate, instead of (6), to use the computational formulas

$$\widetilde{y}_{k+1} = \widetilde{y}_k \oplus h \otimes \sum_{j=0}^{m-1} \alpha_j \otimes \nabla^j f_k, \tag{8}$$

$$f_k := f(t_k, \widetilde{y}_k) \in R^n, \quad \nabla^0 f_k := f_k. \quad \nabla^{j+1} f_k = \nabla^j f_k - \nabla^j f_{k-1}.$$

Instead of the coefficients c_ν from (6) that have quite complicated structure, in (8), the coefficients α_j are computed using simple recurrent relations. To obtain the estimates of errors of procedure

$$\rho_k = h^{m+1} \otimes \alpha_m \otimes y^{(m+1)}([t-(m-1)h, t])$$

it is necessary to compute the Taylor coefficients up to required order. The automation of this process is obtained by using the technique described by Moore [2] of recurrent computation of the Taylor series coefficients. This technique is used in [3]. By the above method, some examples were computed using PASCAL-XSC. This programming language developed at AMI of Karlsruhe University (Germany), presents wide possibilities for correct solving various problems of science and technology. Optimal arithmetical operations are defined in it, with directed roundings # < (to the nearest lesser value), # > (to the nearest greater value); these roundings are necessary performed when obtaining values of the upper and lower functions according to (6), (7) or (8). Inherent to PASCAL-XSC facilities are the constructing modular programs using

dynamic arrays, access to subarrays, optimal dot product, computing expressions with high accuracy and other. To illustrate some of listed facilities, we give the fragments of PASCAL-XSC program on the method described above.

```
program TWO_ADAMS(input,output);
use i_ari,mv_ari,mvi_ari;
var m,n: integer;
function F_rT(t:real;Y:rvector):rvector [0..n-1];
{Computing the vector of right-hand side of the ODE system}
begin ... end;
function Koeff_A(m:integer):rvector[0..m-1];
{The algorithm of the recurrent computation}
{of the coefficient vector α_j in (8);}
     var  G   :rvector[0..m-1];
          j,l :integer;
          s   :real;
begin
     G[0]:=1;
     for j:=1 to m-1 do begin
          s:=0;
          for l:=0 to j-1 do begin
               s:=s+G[l]/(j+1-l);
          end;
     G[j]:=1-s;
     end;
     Koeff_A:=G;
end;
  ..........................................................
procedure haupt(n:integer;m:integer);
var
     i,j,k,l          :integer;
     h,t,tm           :real;
     DeltaT           :interval;
     Y,Yl,Yr          :rvector[0..n-1];
```

```
        Hq                  :rvector[0..n-1];
        MnullQ_l, MQ_l      :rmatrix[0..n-1,0..m-1];
        MnullQ_r, MQ_r      :rmatrix[0..n-1,0..m-1];

  .....
begin read(Y);read(h);read(DeltaT);read(Hq);
        t:=inf(DeltaT);tm:=t;Yl:=Y;Yr:=Y;
        for k:=0 to m-1 do begin          { Computing of}
              MnullQ_l[*,k]:=F_rT(tm,Y); { differences}
              tm:=tm-h; ..... end; .....
        repeat

  .....
          Yl:=Yl+h*MQ_l*Koeff_A(m)-Hq; { Computing of lower and}
          Yr:=Yr+h*MQ_r*Koeff_A(m)+Hq; { upper functions}
          t:=t+h;

  .....
        until t >= sup(DeltaT);
end;
begin
        read(n);read(m);
        haupt(n,m);
end.
```

In this program, the built-in types of data `rvector` and `rmatrix` are used, that imported by means of `use clause`.

One can define and use personal dynamic data types. for example,

```
type vector=dynamic array[*] of real;
     matrix=dynamic array[*,*] of real.
```

Remark. The value of lower and upper functions can be obtained also using interval vector-matrix arithmetic. To do this, it is necessary to modify properly some parts of the program.

In conclusion, we observe that using PASCAL-XSC at Computer center and Mechanics and Mathematics faculty of the Saratov State University begun on February, 1992. The seminars of Prof. Dr. J. Wolff von Gudenberg and Dr. J. Schulze preceded this using, as well as the materials on description of the language and PASCAL-XSC precompiler

for Microsoft C, used for teaching students of the M/M faculty, received from Prof. Dr. Ch.Ullrich and A.G.Yakovlev.

References

1. Filippov, A. F. *Obtaining lower and upper estimates for a solution of differential equations.* In: "Proceedings of the seminar on interval mathematics, Saratov, May 29-31, 1990", pp. 105–110 (in Russian).

2. Moore, R. E. *Interval analysis.* Prentice-Hall, Englewood Cliffs, NJ, 1966.

3. Zyuzin, V. S. *Solving ordinary differential equations by means of Taylor series.* In: "Proc. All-Union conf. on actual problems of applied mathematics, Saratov, May 20–22, 1991", pp. 334–340 (in Russian).

4. Ermakov, O. B. *Double-sided method for solving systems of ordinary differential equations with nondiagonal monotonicity of the right-hand sides.* In: "International symposium on computer arithmetic, scientific computations and mathematical modelling, Albena, September 23–28, 1990".

5. Klatte, R., Kulisch, U., Neaga, M., Ratz, D. and Ullrich, Ch. *PASCAL-XSC, Sprachbeschreibung mit Biespielen.* Springer Verlag, 1991.