1 Solver BiMD

1.1 General information

Authors:	C. Magherini and L. Brugnano
first version:	October, 2005
last update:	September, 2006
language:	Fortran 77
availability:	the code BiMD is freely available (in the public domain)
official link:	http://www.math.unifi.it/~brugnano/BiM/index.html
problem type:	ODEs, DAEs up to index 3
IVPtestset files:	solver: bimd.f
	driver: bimdd.f
	auxiliary files: bimda.f (auxiliary routines)

1.2 Numerical method

The code BiMD (written in FORTRAN 77) is based on Blended Implicit Methods of orders 4, 6, 8, 10 and 12. These are a class of L-stable Block Implicit Methods defined as a suitable combination (*blending*) of two equivalent forms of a basic method in order to favorably meet implementation requirements [BT01, BM02, BMM06, Mag04].

1.3 Implementation details

Nonlinear systems are solved by means of an iterative procedure, called *blended iteration*, based on a nonlinear splitting "naturally" associated to the methods. The strategies for the variation of both the stepsize of integration and the order of the method rely on an estimate of the local truncation errors, obtained through a deferred correction-like procedure, and on an estimate of the convergence properties of the blended iteration. Almost all the details concerning the construction of the code are described in [BM04, BM05, BM06]. The style used during the formulation of the code is very similar to the one used in the codes RADAU and GAM, from which the authors imported some subroutines and comments. Moreover, the name and the meaning of a number of input parameters and local variables have been fully inherited from such codes.

1.4 How to solve test problems with BiMD

Compiling

```
f90  -o dotest  bimdd.f  problem.f  bimda.f  bimd.f  report.f,
```

will yield an executable `dotest` that solves the problem, of which the Fortran routines in the format described in Section IV.3 are in the file `problem.f`.

As an example, we perform a test run, in which we solve problem HIRES. Figure I.1.1 shows what one has to do.

References

[BM02] L. Brugnano and C. Magherini. Blended implementation of block implicit methods for odes. *Appl. Numer. Math.*, 42:19–45, 2002.

[BM04] L. Brugnano and C. Magherini. The bim code for the numerical solution of odes. *J. Comput. Appl. Math.*, 164-165:145–158, 2004.

```
$ f90 -O5 -o dotest bimdd.f hires.f bimda.f  bimd.f report.f
$ ./dotest

  Test Set for IVP Solvers (release 2.3)

  Solving Problem HIRES using BiMD

 User input:

 give relative error tolerance:
1d-4
 give absolute error tolerance:
1d-4
 give initial stepsize:
1d-4

 Numerical solution:

                                                     scd
       solution component            ---------------------------    ignore
                                     mixed       abs        rel    mix - abs,rel
 ----------------------------------  -----      -----      -----   -------------
 y(  1) =  0.7370390869868378E-003    7.04       7.04       3.90
 y(  2) =  0.1442309432867305E-003    7.75       7.75       3.91
 y(  3) =  0.5886726446999230E-004    7.70       7.70       3.47
 y(  4) =  0.1175514405948053E-002    6.86       6.86       3.93
 y(  5) =  0.2382225270095926E-002    5.38       5.38       2.76
 y(  6) =  0.6222129415035646E-002    4.78       4.77       2.57
 y(  7) =  0.2849350956905541E-002    6.19       6.19       3.64
 y(  8) =  0.2850649043094471E-002    6.19       6.19       3.64

 used components for  scd                8          8          8
 scd of Y (maximum norm)              4.78       4.77       2.57


 using mixed error yields mescd       4.78
 using relative error yields scd                            2.57

 Integration characteristics:

    number of integration steps        36
    number of accepted steps           33
    number of f evaluations           559
    number of Jacobian evaluations     30
    number of LU decompositions        36

 CPU-time used:                       0.0020 sec
```

FIGURE I.1.1: *Example of performing a test run, in which we solve problem HIRES with BiMD. The experiment was done on an ALPHAserver DS20E, with a 667MHz EV67 processor. We used the Fortran 90 compiler* `f90` *with the optimization flag* `-O5`.

[BM05] L. Brugnano and C. Magherini. Some linear algebra issues concerning the implementation of blended implicit methods. *Numer. Linear Alg. Appl.*, 12:305–314, 2005.

[BM06] L. Brugnano and C. Magherini. Economical error estimates for block implicit methods for odes via deferred correction. *Appl. Numer. Math.*, 56:608–617, 2006.

[BMM06] L. Brugnano, C. Magherini, and F. Mugnai. Blended implicit methods for the numerical solution of dae problems. *J. Comput. Appl. Math.*, 189:34–50, 2006.

[BT01] L. Brugnano and D. Trigiante. Block implicit methods for odes. *Recent Trends in Numerical Analysis, D. Trigiante Ed. Nova Science Publ. Inc.*, pages 81–105, 2001.

[Mag04] C. Magherini. *Numerical Solution of Stiff ODE-IVPs via Blended Implicit Methods: Theory and Numerics.* PhD thesis, Dipartimento di Matematica U. Dini, Università degli Studi di Firenze, September 2004.