Comparing Routines for the Numerical Solution of Initial Value Problems of Ordinary Differential Equations in Multiple Shooting


Received July 26, 1976

Summary. The numerical solution of two-point boundary value problems and problems of optimal control by shooting techniques requires integration routines. By solving 15 real-life problems four well-known integrators are compared relative to reliability, fastness and precision. Hints are given, which routines could be used for a problem.

1. Introduction

In recent years a series of refined numerical methods has been developed for the solution of initial value problems in ordinary differential equations. The large number of published integration routines necessitates a comparison of all these methods. Enright et al. [16], Hull [23], and Davenport et al. [10] have tested and compared integration routines by solving carefully selected initial value problems.

In applications many nonlinear two-point boundary value problems and problems of optimal control arise. Convenient methods for the numerical solution of these problems are shooting algorithms. Since these require a repeated computation of initial value problems, integration routines also play an important role.

In this paper the authors have tested the suitability and the behaviour of certain integrators for a multiple shooting method. They are mainly interested in a comparison of integration routines, which help a user select a method for a particular problem. Secondly several boundary value problems and optimal control problems are proposed and treated, which are realistic and representative of real-life applications. This may stimulate further comparisons of algorithms for solving boundary value problems.

In order to compare the integrators, 15 two-point boundary value problems are solved, which arise in different application areas: mathematics, physics, engineering, space science and economics. This report is concerned only with non-stiff problems. All the selected examples are solved by means of a multiple shooting technique described, for example, in Stoer and Bulirsch [40]. In this algorithm a modification of Newton's method due to Deuflhard [11] is used.

It may be recalled that in multiple shooting, sequences of initial value problems must be solved with iteratively determined initial values (for realistic prob-
lems about 100 initial value problems in every iteration involving numerical differentiation). Because of the artificial character of these initial values the integration routines have to overcome harder difficulties than in the solution of natural initial value problems. In addition, the singularities of the solutions of nonlinear ordinary differential equations depend on the initial values (movable singularities). That is why the integration routines must be reliable, fast and precise.

2. Integration Routines

The results of the investigations of 20 routines in [16] suggested the following four integrators:

DIFSY 1: Bulirsch-Gragg-Stoer extrapolation method.
This is an extrapolation algorithm based on the midpoint rule with Gragg stabilization [21]. It was first published by Bulirsch and Stoer [6]. In Hussels [24] an improved stepsize control was implemented.

VOAS: Adams-Sedgwick method of variable order and variable step.
This is an implementation of a variable order variable step Adam's method developed by Sedgwick [39].

RKF7: Runge-Kutta-Fehlberg method of seventh order,
RKF4: Runge-Kutta-Fehlberg method of fourth order.
These are Runge-Kutta methods with built-in estimators for the local error, see [16]. They are based on the formulas of orders seven and four, respectively, developed by Fehlberg [18, 19] and England [17].

3. Multiple Shooting Method

A detailed description of the multiple shooting method for the solution of a two-point boundary value problem

\[ y' = f(t, y); \quad y: [a, b] \to \mathbb{R}^N, \quad f: [a, b] \times \mathbb{R}^N \to \mathbb{R}^N, \]

\[ r(y(a), y(b)) = 0; \quad r: \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N \]

may be found e.g. in [40] and [7].

Here the interval \([a, b]\) is suitably subdivided

\[ a = t_1 < t_2 \ldots < t_{M-1} < t_M = b \quad (M \text{ nodes}). \]

Denoting \( y(t; t_j, s_j), \quad j = 1, \ldots, M-1 \) the solution of the initial value problem

\[ y' = f(t, y); \quad y(t_j) = s_j, \quad t \in [t_j, t_{j+1}] \]

the \( N \)-vectors \( s_j \) have to be determined so, that the following \( N(M-1) \) conditions hold:

continuity conditions (for \( M > 2 \))

\[ F_j(s_j, s_{j+1}) := y(t_{j+1}; t_j, s_j) - s_{j+1} = 0 \quad j = 1, \ldots, M - 2 \]
boundary conditions

\[ F_{M-1}(s_1, s_{M-1}) := r(s_1, y(t_M; t_{M-1}, s_{M-1})) = 0. \]  

(2.2.b)

The conditions (2.2) define a system of \( N(M - 1) \) nonlinear equations

\[ F(s) := \begin{bmatrix} F_1(s_1, s_2) \\ \vdots \\ F_{M-2}(s_{M-2}, s_{M-1}) \\ F_{M-1}(s_1, s_{M-1}) \end{bmatrix} = 0 \]  

with \( s := \begin{bmatrix} s_1 \\ \vdots \\ s_{M-1} \end{bmatrix} \).  

(2.3)

This system is solved numerically by the modified Newton-method:

\[ s^{k+1} = s^k + \lambda_k A s^k \quad 0 < \lambda_k \leq 1, \]  

(2.4.a)

\[ A s^k = -DF(s^k)^{-1} F(s^k) \]  

(2.4.b)

\( DF(s) \) denotes the Jacobian matrix.

A good strategy for choosing the \( \lambda_k \) has been developed by Deuflhard [12]. Starting the process (2.4) the following initial data must be available

\[ s^{0}_j := \begin{bmatrix} s^0_{1j} \\ \vdots \\ s^0_{Nj} \end{bmatrix} \quad j = 1, \ldots, M - 1. \]

With the following abbreviations

\[ G_j := \frac{\partial y(t_{j+1}; t_j, s_j)}{\partial s_j} \quad j = 1, \ldots, M - 1, \]

\[ A := \frac{\partial r}{\partial y(a)} \quad B := \frac{\partial r}{\partial y(b)}. \]

(2.4.b) can be written in more detail (the \( k \)-index is omitted):

\[ \begin{bmatrix} G_1 & -I \\ G_2 & -I \\ \vdots & \vdots \\ 0 & G_{M-2} & -I \end{bmatrix} \begin{bmatrix} A s_1 \\ A s_2 \\ \vdots \\ A s_{M-2} \\ A s_{M-1} \end{bmatrix} = - \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_{M-2} \\ F_{M-1} \end{bmatrix}. \]  

(2.4.b')

The \( A s_j \) allow a recursive determination by

\[ A s_1 = E^{-1} \omega, \]  

(2.5.a)

\[ A s_j = G_{j-1} A s_{j-1} + F_{j-1} \quad j = 2, \ldots, M - 1 \]  

(2.5.b)

with

\[ E := A + B G_{M-1} \]  

iteration matrix.

(2.6.a)

\[ \omega := -(F_{M-1} + B G_{M-1} F_{M-2} + \cdots + B G_{M-1} G_1 F_1). \]  

(2.6.b)

\( A, B \) and the \( G_j \) are computed by numerical differentiation. This requires the calculation of \( N \) trajectories. The use of Broyden's approximation formulas [3] reduces computing time up to 30%. It is apparent that the choice of a reliable, fast and precise integration routine plays an important role in multiple shooting.
Remark. The piecewise constructed function
\[ y(t) := y(t; t_j, s_j) \quad \text{for} \quad t \in [t_j, t_{j+1}) \quad j = 1, \ldots, M - 1 \]
is called a trajectory. In particular for \( s_j = s_j^0 \)
y(t) is the starting trajectory.

4. Numerical Examples

To compare the four integration methods, problems were chosen from the
areas of mathematics, physics, technical science, economics and space science.
These examples include boundary value problems, integral equations, eigenvalue
problems and problems of optimal control. The problems range from simple ones
of theoretical nature to highly complex ones.

All examples were transformed into two-point boundary value problems (ex-
amples 1–11) or into boundary value problems with switching functions (examples
12–15). They were solved by the standard multiple shooting algorithm BOUND-
SOL and its modified version especially trimmed to handle optimal control
problems.

The experiments were run on the TR440 of the Leibniz-Rechenzentrum der
Bayerischen Akademie der Wissenschaften. The computations were performed
in FORTRAN single precision with a 38 bit mantissa (examples 1–11) and in
FORTRAN double precision with an 84 bit mantissa (examples 12–15).

The statistics of the comparisons include the following parameters:

a) Parameters concerning the problem:

N: Number of differential equations.

M: Number of nodes for the multiple shooting method.

NT: Number of trajectories.

TOL: Tolerance for the initial value methods.

OVHB: Overhead time for BOUNDSOL, i.e. the difference between the total
time and the time spent in solving initial value problems.

EPS: Relative tolerance for BOUNDSOL (\( 1 \times 10^{-6} \) for all examples).

b) Parameters concerning the comparison of the methods:

TIME: Total time to solve the problem.

OVHM: Overhead time for METHOD, i.e. the difference between the time
solving the initial value problems and the time spent in evaluating the
functions.

NFC: Number of function calls. NFC depends on the number and position
of the nodes. A proper choice of the nodes, depending on a special inte-
gration routine, can produce a 30% reduction in computing time. An
optimal choice for all four routines is not possible because of the dif-
ferent kinds of stepsize control.
c) Parameter concerning the final assessment:

NCT: Exact time of solving an example by one method in per cent of the average time of the four methods.

Because of the different length of the problems a final assessment of the methods and their properties is difficult. To overcome this difficulty, "a normed computing time" NCT is introduced. This simply calculated number has the advantage of being independent of the computer used, and of the computing time. For the single examples the NCT is not important, it is only used for the final comparison in the conclusion (see Chapter 5).

Example 1. A standard test problem \((N=2)\).

This linear two-point boundary value problem is discussed in \([40]\), p. 205 (see also Pereyra \([30]\) and Daniel and Martin \([9]\)):

\[
-y'' + 400y = -400 \cos^2(\pi t) - 2\pi^2 \cos(2\pi t) \\
y(0) = y(1) = 0.
\]

Single shooting technique will fail to solve this problem \([40]\). Since the solution has one exponentially increasing and one exponentially decreasing component, the following transformation is introduced:

\[
y_1 = y \\
y_2 = 20y + y'.
\]

Nodes: 0, 0.25, 0.5, 0.75, 1.

Zero initial data

<table>
<thead>
<tr>
<th>Method</th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
<th>M</th>
<th>NT</th>
<th>TOL</th>
<th>OVHB</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>2.5</td>
<td>1.4</td>
<td>3901</td>
<td>39</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOAS</td>
<td>4.6</td>
<td>3.7</td>
<td>2222</td>
<td>72</td>
<td></td>
<td></td>
<td>1.10</td>
<td></td>
</tr>
<tr>
<td>RKF7</td>
<td>3.2</td>
<td>1.8</td>
<td>4684</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td>0.3</td>
</tr>
<tr>
<td>RKF4</td>
<td>15.2</td>
<td>8.7</td>
<td>22635</td>
<td>238</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example 2. A singular bifurcation problem \((N=2)\).

\[-u'' + [(u^3)']' = \lambda u, \quad u(0) = u(1) = 0.\]

This problem has been studied by Küpper \([27]\). It approximates the partial differential equation of the model of a continuous vibrating string considered by Fermi et al. \([20]\). As discussed in \([27]\), for \(\lambda \in [8, 2\pi]\) the problem has a positive solution which is symmetric and bifurcates from the trivial solution at \(\lambda = \pi^2\). Outside of \([8, 2\pi]\) the problem has no positive solution.

The solution at \(\lambda = 8\) is calculated. For this parameter value the problem possesses a numerical singularity at \(t=0\). For that reason a series approximation in the first subinterval is used.
Nodes: equidistant
Initial data: solution at $\lambda = 8.5$ (obtained by continuation)

<table>
<thead>
<tr>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>1.2</td>
<td>0.6</td>
<td>2023</td>
</tr>
<tr>
<td>VOAS</td>
<td>2.2</td>
<td>1.6</td>
<td>1667</td>
</tr>
<tr>
<td>RKF7</td>
<td>1.6</td>
<td>0.9</td>
<td>2704</td>
</tr>
<tr>
<td>RKF4</td>
<td>1.0</td>
<td>0.4</td>
<td>1300</td>
</tr>
</tbody>
</table>

Example 3. An example due to Troesch ($N=3$).

The two-point boundary value problem is given by:

$$y'' = \lambda \sinh (\lambda y), \quad \lambda \geq 0,$$
$$y(0) = 0, \quad y(1) = 1.$$  

This problem had been discussed by several authors (e.g. Scott and Watts [38], p. 74/75, where further references are listed). It describes the confinement of a plasma column by radiation pressure. The exact solution of the problem in terms of Jacobian elliptic functions can be found in [40], p. 169. For the numerical evaluation of the Jacobian elliptic functions it is recommended to employ the computationally economic, rapidly convergent method of the arithmetic-geometric mean [4]. Multiple shooting was used as a background procedure for comparison purposes only.

The problem was solved by a modified continuation method [13], a trivial differential equation and a linear boundary condition for the homotopy parameter $\lambda$ is added.

The sensitivity of the problem is caused by a logarithmic singularity at $t=t_\lambda > 1$ of the solution of the associated initial value problem. One might expect numerical difficulties for the multiple shooting method because $t_\lambda$ depends on the parameter $\lambda$ and approaches $1^+$ as $\lambda$ increases. Indeed, the norm of the iteration matrix increased from $1.100$ to $1.109$ as $\lambda$ varied from 1 to 17.5. Therefore single precision computation had to be terminated.

Nodes: 0, 0.3, 0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.92, 0.94, 0.96, 0.97, 0.98, 0.99, 1.

Initial data: solution at $\lambda = 7.25$.

<table>
<thead>
<tr>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>3.5</td>
<td>1.7</td>
<td>3393</td>
</tr>
<tr>
<td>VOAS</td>
<td>11.4</td>
<td>9.0</td>
<td>5224</td>
</tr>
<tr>
<td>RKF7</td>
<td>8.9</td>
<td>5.3</td>
<td>9114</td>
</tr>
<tr>
<td>RKF4</td>
<td>5.5</td>
<td>3.2</td>
<td>5212</td>
</tr>
</tbody>
</table>

Remark. For small values of $\lambda$ VOAS or Runge-Kutta methods are faster than DIFSY1, but for all $\lambda \geq 4$ DIFSY1 turns out to be best. First of all, VOAS fails at $\lambda = 9.75$ then RKF7 at $\lambda = 11.5$ and RKF4 at $\lambda = 14.0$. The standard multiple shooting algorithm suggests a new node be inserted which coincides with 1. in the first four digits. The last successful step for DIFSY1 is $\lambda = 17.5$. 
Example 4. Artificial boundary layer problem \((N = 2)\).

This linear problem was treated in Daniel and Martin [9], Lentini and Pereyra [28], and Pereyra [30]:

\[
y'' = - \frac{3 \lambda y}{(\lambda + 1)^3}; \quad y(0) = 0, \quad y(0.1) = \frac{0.1}{\sqrt{\lambda + 0.01}}.
\]

At \(t = 0\) the problem has a boundary layer of thickness \(\sqrt{\lambda}\). One expects serious difficulties for the stepsize control of the four routines as \(\lambda\) approaches zero.

Nodes: 0, \(\frac{1}{10^7}, 0.01, 0.1\).
Initial data: solution at \(\lambda = 1 \cdot 10^{-5}\).

<table>
<thead>
<tr>
<th></th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>0.7</td>
<td>0.4</td>
<td>1464</td>
<td>25</td>
</tr>
<tr>
<td>VOAS</td>
<td>7.0</td>
<td>6.6</td>
<td>3615</td>
<td>222</td>
</tr>
<tr>
<td>RKF7</td>
<td>1.4</td>
<td>1.0</td>
<td>2751</td>
<td>44</td>
</tr>
<tr>
<td>RKF4</td>
<td>3.4</td>
<td>3.1</td>
<td>7116</td>
<td>108</td>
</tr>
</tbody>
</table>

Remark. Solving this example DIFSY1 is the most reliable and VOAS the most susceptible routine. The last successful steps has been for

VOAS: \(\lambda = 1 \cdot 10^{-8}\)
RKF4, RKF7: \(\lambda = 1 \cdot 10^{-10}\)
DIFSY1: \(\lambda = 1 \cdot 10^{-13}\)

The stepsize control of the routines orders a new node which is too close to a given one.

Example 5. An integral equation of the second kind \((N = 5)\).

\[
6 \pi \varphi(s) = \int_0^n (3 \sin \tau \sin s + 2 \sin 2s \sin 2\tau) (\varphi(\tau) + \varphi^3(\tau)) d\tau.
\]

This equation has three different solutions in addition to the trivial solution (Pimbley [32]). Each of them can be obtained by multiple shooting, after transforming the integral equation into a boundary value problem by:

\[
\begin{align*}
y_1(t) &= \varphi(t), \\
y_2(t) &= \int_0^t \sin \tau (\varphi(\tau) + \varphi^3(\tau)) d\tau, \\
y_3(t) &= \int_t^n \sin \tau (\varphi(\tau) + \varphi^3(\tau)) d\tau, \\
y_4(t) &= \int_0^t \sin (2\tau) (\varphi(\tau) + \varphi^3(\tau)) d\tau, \\
y_5(t) &= \int_t^n \sin (2\tau) (\varphi(\tau) + \varphi^3(\tau)) d\tau.
\end{align*}
\]
Nodes: $0, 0.5, \pi/2, 2, \pi$.
Initial data: $y \equiv 2$ except for the given boundary values.
Using the chosen data, multiple shooting yields the only positive solution.

<table>
<thead>
<tr>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
<th>M</th>
<th>NT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>7.5</td>
<td>3.5</td>
<td>5361</td>
<td>96</td>
<td>M = 5</td>
</tr>
<tr>
<td>VOAS</td>
<td>10.9</td>
<td>8.0</td>
<td>3411</td>
<td>140</td>
<td>NT = 18</td>
</tr>
<tr>
<td>RKF7</td>
<td>5.8</td>
<td>3.0</td>
<td>3471</td>
<td>74</td>
<td>TOL = $1 \times 10^{-6}$</td>
</tr>
<tr>
<td>RKF4</td>
<td>7.0</td>
<td>3.6</td>
<td>4379</td>
<td>90</td>
<td>OVHB = 0.8</td>
</tr>
</tbody>
</table>

**Example 6. Membrane theory ($N = 2$).**

This problem arises in the theory of stress distribution in a spherical membrane having normal and tangential loads (see Lentini and Pereyra [28], Russell and Shampine [36], and Scott [37]). The following differential equation holds:

$$y'' + (3 \cot (t) + 2 \tan (t)) y' + 0.7y = 0,$$

$$y(30^\circ) = 0 \quad y(60^\circ) = 5.$$  

The sharp rise in $y(t)$ and its derivatives may cause difficulties. The problem is scaled by a factor $1 \times 10^{-4}$.

Nodes: $30^\circ, 31^\circ, 60^\circ$.
Initial data: $y \equiv 0, y' \equiv \pi/1800$.

<table>
<thead>
<tr>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
<th>M</th>
<th>NT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>7.9</td>
<td>4.4</td>
<td>15615</td>
<td>138</td>
<td>M = 3</td>
</tr>
<tr>
<td>VOAS</td>
<td>5.3</td>
<td>4.5</td>
<td>3097</td>
<td>93</td>
<td>NT = 6</td>
</tr>
<tr>
<td>RKF7</td>
<td>4.3</td>
<td>2.6</td>
<td>6742</td>
<td>75</td>
<td>TOL = $1 \times 10^{-6}$</td>
</tr>
<tr>
<td>RKF4</td>
<td>5.4</td>
<td>3.4</td>
<td>8796</td>
<td>94</td>
<td>OVHB = 0.2</td>
</tr>
</tbody>
</table>

**Example 7. Heat conduction ($N = 3$).**

The equations for one-dimensional heat conduction with nonlinear heat generation are treated in Na and Tang [29] and Stoer and Bulirsch [40]:

$$y'' = - \frac{y'}{t} - \lambda y^\gamma,$$

$$y'(0) = y(1) = 0$$

where $\lambda$: heat generation constant $0 < \lambda \leq 0.8,$

$y$: temperature distribution.

The problem possesses a numerical singularity at $t=0$. In actual computation the differential equation must be separated (see [40] and [34]).
Example 8. Lubrication theory \((N = 2)\).

The following nonlinear eigenvalue problem was treated by Cole et al. [8] and Keller [25]. It arises in the theory of lubrication and concerns the flow of a viscous compressible fluid through a very narrow gap.

\[
y' = \frac{1}{\varepsilon} \left( \sin^2 t - \lambda \frac{\sin^2 t}{y} \right); \quad -\pi/2 \leq t \leq \pi/2
\]

\[
y(\pi/2) = y(-\pi/2) = 1;
\]

\(\varepsilon\) is a parameter, \(\lambda\) the required eigenvalue.

This boundary value problem was solved numerically for \(\varepsilon = 1/100\). For this parameter value the solution \(y\) approaches zero at \(t = 0\).

The single shooting method was performed in the backward direction (integrating from \(\pi/2\) to \(-\pi/2\)). In this case the norm of the iteration matrix was about \(10^{-10}\). On the other hand, during forward integration the norm increased to \(0.5\times 10^{-10}\), indicating high sensitivity of the problem.

Nodes: \(-\pi/2, \pi/2\) \((M = 2, \text{single shooting})\).

Initial data: \(y = 1, \lambda = 4/3\).

Example 9. Nonsymmetric bending of elliptic cylindrical tubes \((N = 6)\).

Recently Weinitschke [43, 44] studied the nonlinear problem of bending a thin-walled cylindrical tube with elliptic cross section. No assumptions are made of bending the tube about the minor or major axis. The differential equations are:

\[
\beta'' = \alpha^2 f \sqrt{\cos^2 \xi + \varepsilon^2 \sin^2 \xi} \left( \sin \beta (\cos \xi + \gamma \varepsilon \sin \xi) - \cos \beta (\gamma \cos \xi - \varepsilon \sin \xi) \right)
\]

\[
- \frac{g \sin \xi \cos \xi}{\cos^2 \xi + \varepsilon^2 \sin^2 \xi} \beta',
\]

\[
f'' = \sqrt{\cos^2 \xi + \varepsilon^2 \sin^2 \xi} \left( \cos \beta (\cos \xi + \gamma \varepsilon \sin \xi) + \sin \beta (\gamma \cos \xi - \varepsilon \sin \xi) \right)
\]

\[
- \frac{g \sin \xi \cos \xi}{\cos^2 \xi + \varepsilon^2 \sin^2 \xi} f'.
\]
where:

\[ \xi : \text{angle } 0 \leq \xi \leq 2\pi, \]
\[ f(\xi) : \text{dimensionless stress function}, \]
\[ \beta(\xi) : \text{dimensionless angular deflection}, \]
\[ b : \text{minor axis}, \]
\[ a : \text{major axis}, \]
\[ R_x : \text{projection of the deformed axis to the } x-z \text{ plane, assumed to be radius of a circle}, \]
\[ R_y : \text{projection to the } y-z \text{ plane}, \]
\[ D : \text{flexural rigidity}, \]
\[ 1/A : \text{extensional stiffness}, \]
\[ e = \frac{b}{a}, \quad q = 1 - e^2, \]
\[ \gamma = \frac{R_x}{R_y}, \quad \alpha = \frac{a^2}{\sqrt{AD}R_x}. \]

Symmetry of the problem leads to the boundary conditions:

\[ \beta(0) = \beta\left(\frac{\pi}{2}\right) = \arctan \gamma, \]
\[ f'(0) = 0, \quad f\left(\frac{\pi}{2}\right) = 0. \]

Additionally, two integrals for the moments \( m_x, m_y \) are computed:

\[ m_x = -\frac{2\alpha}{\pi} \int_0^{\pi/2} f\left[\left((1+C)\cos\beta + S\sin\beta\right)\cos t - \epsilon\left(1+C\sin\beta - S\cos\beta\right)\sin t\right] dt, \]
\[ m_y = -\frac{2\alpha}{\pi} \int_0^{\pi/2} f\left[\left((1-C)\sin\beta + S\cos\beta\right)\cos t + \epsilon\left(1-C\cos\beta - S\sin\beta\right)\sin t\right] dt \]

where: \( S = \sin(2\arctan \gamma), C = \cos(2\arctan \gamma). \)

In the simpler case of a cylindrical tube, \( e = 1 \), solutions have been obtained by means of a perturbation technique by Reissner and Weinitschke [33]. Quasi-linearization has been successfully applied by Thurston [42].

The problem was solved for the parameters:

\[ e = 0.9, \quad \gamma = 0.25, \quad \alpha = 0.5. \]

Nodes: 0, \( \pi/4 \), \( \pi/2 \).

Initial data:

\[ \beta = \arctan \gamma \]
\[ \beta' = \pi^2/30 \quad \text{estimated from [33]}, \]
\[ f(0) = -1, \quad f\left(\frac{\pi}{4}\right) = -0.5, \quad f\left(\frac{\pi}{2}\right) = 0. \]
\[ f'(0) = 0, \quad f'\left(\frac{\pi}{4}\right) = 2/\pi, \quad f'\left(\frac{\pi}{2}\right) = 2/\pi, \]
\[ m_x' = 0, \]
\[ m_y' = 0. \]
Example 10. A hydrodynamic problem ($N = 5$).

The problem describes a laminar boundary layer produced by the rotating flow of a viscous incompressible fluid over a stationary infinite disk. The external velocity should vary as some power of the radius: $v \sim r^{-n}$; further the fluid is assumed to be interacting with a magnetic field represented by the parameter $s$. A complete discussion of this problem can be found in King and Lewellen [26].

The original version leads to a two-point boundary value problem:

$$
\psi'' + \left( \frac{3-n}{2} \right) \psi' + n (\psi')^2 - 1 + \Phi^2 - s \psi' = 0,
$$

$$
\Phi'' + \left( \frac{3-n}{2} \right) \psi' + (n-1) \psi' \Phi - s (\Phi - 1) = 0
$$

boundary conditions:

$$
\psi(0) = \psi'(0) = \Phi(0) = 0,
$$

$$
\psi'(t_f) = 0, \quad \Phi(t_f) = 1
$$

where $n$ and $s$ are given parameters.

For several parameters $n, s, t_f$ numerical solutions of this problem were obtained by Holt [22] using a combination of quasilinearization and finite difference methods and by Roberts and Shipman [35] using a shooting method.

In [35] the solution is given for the parameters

$$
n = -0.1, \quad s = 0.2, \quad t_f = 11.3.
$$

Nodes: equidistant.

Initial data: solution at $t_f = 11.3$.

Example 11. Re-entry problem for an Apollo orbiter type vehicle ($N = 7$).

Of all the many difficult problems due to manned interplanetary travel perhaps the most critical is that of returning the vehicle from outer space to the earth’s surface. The chief problem with aerodynamic breaking is associated with the severe heating effects experienced by the vehicle. Therefore reentry trajectories are searched which minimize this quantity. This optimal control problem
leads to seven differential equations involving the adjoint variables $\lambda_v$, $\lambda_\gamma$, $\lambda_\xi$ and the free total flight time $T$ (cf. [40]).

**Physical differential equations:**

\[
\begin{align*}
\dot{v} &= \left[ \frac{S_0 v^2}{2m} C_w(u) - \frac{g \sin \gamma}{(1 + \xi)^2} \right] \cdot T \quad \text{(velocity)}, \\
\dot{\gamma} &= \left[ \frac{S_0 v^2}{2m} C_A(u) + \frac{v \cos \gamma}{R(1 + \xi)} - \frac{g \cos \gamma}{v(1 + \xi)^2} \right] \cdot T \quad \text{(flight path angle)}, \\
\dot{\xi} &= \frac{v \sin \gamma}{R} \cdot T \quad \text{(normalized altitude)}.
\end{align*}
\]

**Hamiltonian:**

\[
H = 10 v^3 \sqrt{\theta + (\lambda_v \dot{v} + \lambda_\gamma \dot{\gamma} + \lambda_\xi \dot{\xi})} / T
\]

where

\[
\begin{align*}
C_w(u) &= 1.174 - 0.9 \cos u, \\
C_A(u) &= 0.6 \sin u \quad (u \text{ is the control variable}), \\
\sin u &= -0.6 \lambda_\gamma / \alpha, \\
\cos u &= -0.9 v \lambda_\gamma / \alpha, \\
\alpha &= \sqrt{(0.6 \lambda_\gamma)^2 + (0.9 v \lambda_\gamma)^2}, \\
\theta &= \theta_0 e^{-\beta R \xi}, \\
R &= 209, \quad \beta = 4.26, \quad \theta_0 = 2.70410^{-3}, \quad g = 3.217210^{-4}, \\
S/m &= 53 200.
\end{align*}
\]

The boundary conditions are given by:

\[
\begin{align*}
v(0) &= 0.36, \quad v(1) = 0.27, \\
\gamma(0) &= -8.1 \pi / 180, \quad \gamma(1) = 0, \\
\xi(0) &= 4/R, \quad \xi(1) = 2.5/R, \\
H \big|_{\xi = 1} &= 0.
\end{align*}
\]

**Nodes: Initial data:**

<table>
<thead>
<tr>
<th>$\nu$</th>
<th>$\lambda_v$</th>
<th>$\gamma$</th>
<th>$\lambda_\gamma$</th>
<th>$\xi$</th>
<th>$\lambda_\xi$</th>
<th>$T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>0.2700E+00</td>
<td>0.0</td>
<td>0.119617224888E-01</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.6000E+00</td>
<td>0.47500000000E+00</td>
<td>-0.35100000000E+02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.762</td>
<td>0.2750E+00</td>
<td>0.70685834705E-01</td>
<td>0.11722488038E-01</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.3900E+00</td>
<td>0.31500000000E+00</td>
<td>-0.65700000000E+01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.496</td>
<td>0.2880E+00</td>
<td>0.24085543677E-01</td>
<td>0.10574162680E-01</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.2280E+00</td>
<td>0.12600000000E+00</td>
<td>-0.34600000000E+01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.26</td>
<td>0.3300E+00</td>
<td>-0.21293016874E-01</td>
<td>0.82775119615E-02</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.1070E+00</td>
<td>-0.40000000000E-01</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.097</td>
<td>0.3607E+00</td>
<td>-0.12217304764E+00</td>
<td>0.140191438756E+01</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.5000E-01</td>
<td>-0.55000000000E-01</td>
<td>-0.13000000000E+01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.</td>
<td>0.3600E+00</td>
<td>-0.14137166941E+00</td>
<td>0.19138755981E-01</td>
<td>0.230E+03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.6000E-01</td>
<td>-0.60000000000E-01</td>
<td>-0.20000000000E+00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Remark. In order to prevent an overflow of the right hand side using RKF 7, the argument of the exponential function must be limited.

Example 12. Optimal control of a mass production \((N=3)\).

This optimal control problem is due to Bauer, Neumann [2] and describes an automatic machine producing a mass article:

- \(v\): production velocity: \(v \in [0,1]\),
- \(\alpha v\): articles produced per unit-time: \(\alpha > 0\),
- \(1-x\): percentage of refuse: \(x \in [0,1]\),
- \(\alpha x v\): returns of the production per unit-time,
- \(k(v,t)\): production costs per unit-time: \(k(v,t) = \frac{1}{2} v^2 e^{rt}\).

The profit \(I\) is to be maximized:

\[
I(x,v) = \int_0^T [\alpha x v - k(v,t)] dt
\]

under the restriction (depreciation):

\[
\dot{x} = -bxv, \quad b > 0,
\]

\(x(0) = 1\)

where \(\alpha = 2\), \(b = 0.08\) and \(T = 10\).

The associated two-point boundary value problem is given by

\[
\dot{p} = v \left(\alpha x - \frac{1}{2} v e^{rt}\right), \quad p(0) = 0,
\]

\[
\dot{x} = -bxv, \quad x(0) = 1,
\]

\[
\dot{\lambda}_x = v (\lambda_x b - \alpha), \quad \lambda_x(1) = 0
\]

with the control variable

\[
v = \begin{cases} 
0, & \text{if } \tilde{v} < 0 \text{ (this case does not occur in actual computation)} \\
\tilde{v}, & \text{if } \tilde{v} \in [0,1] \\
1, & \text{if } \tilde{v} > 1,
\end{cases}
\]

\[
\tilde{v} = e^{-\frac{t}{T}} [x(\alpha - \lambda_x b)]
\]

switching function: \(S = \tilde{v} - 1\).
Nodes: equidistant.
Zero initial data except for the given boundary values.

<table>
<thead>
<tr>
<th></th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFS</td>
<td>6.1</td>
<td>1.9</td>
<td>2729</td>
<td>49</td>
</tr>
<tr>
<td>VOAS</td>
<td>7.3</td>
<td>4.6</td>
<td>1420</td>
<td>59</td>
</tr>
<tr>
<td>RKF7</td>
<td>17.0</td>
<td>11.3</td>
<td>4389</td>
<td>137</td>
</tr>
<tr>
<td>RKF4</td>
<td>19.2</td>
<td>12.5</td>
<td>5147</td>
<td>155</td>
</tr>
</tbody>
</table>

Example 13. Optimal economic planning of a growing nation (N = 7).

This example describes the economy of a growing nation which suffers from great unemployment due to the business structure. A policy leading to full employment and stable growth is required. The data used held for Algeria in the year 1961.

The economical model due to Stoleru [41] includes one constrained control variable and one first-order constrained state variable. Using multiple shooting technique Wick [45] has computed the associated two-point boundary value problem. The differential equations which consider two cases are given by (x ∈ [0,1] is the independent variable):

**Case 1.** On unconstrained arcs (g := mσe^{γx}T − z < 0):

\begin{align*}
\dot{y} &= xu \cdot y \cdot T, \\
\dot{z} &= x(1−u)y \cdot T, \\
\dot{\lambda}_y &= −(\lambda_y x u + \lambda_z x(1−u)) \cdot T, \\
\dot{\lambda}_z &= \dot{\lambda}_r = \dot{i}_0 = \ddot{T} = 0.
\end{align*}

**Case 2.** On constrained arcs (g ≡ 0) the differential equations for \( \lambda_y, \lambda_z \) and \( \lambda_r \) change to:

\begin{align*}
\dot{\lambda}_y &= −x \lambda_y \cdot T, \\
\dot{\lambda}_z &= 0, \\
\dot{\lambda}_r &= (\lambda_y − \lambda_z)γ^2 z \cdot T
\end{align*}

with the constants \( α = 0.25, \ σ = 3.1, \ A = B = 4, \ γ = 0.125, \) and \( m = 0.45. \) The adjoint variables \( \lambda_z \) and \( \lambda_r \) are discontinuous at the entry point \( x_1 \) of the constrained arc:

\begin{align*}
\lambda^+_z|_{x=x_1} &= \lambda^-_z|_{x=x_1} \cdot -l_0, \\
\lambda^+_r|_{x=x_1} &= \lambda^-_r|_{x=x_1} \cdot +l_0 m σ γ e^{γx_1}.
\end{align*}

The control variable \( u \) satisfies:

\begin{align*}
u &= \begin{cases}
1, & \text{if } x < x_1 \text{ (entry point)}; \quad g < 0 \\
1 − \frac{γ z}{α y}, & \text{on the constrained arc}; \quad g ≡ 0 \\
0, & \text{if } x > x_2 \text{ (exit point)}; \quad g < 0.
\end{cases}
\end{align*}
The boundary conditions are:
\[
\begin{align*}
y(0) &= 1 \quad y(1) = Ae^{\gamma T}, \\
z(0) &= 0 \quad z(1) = Be^{\gamma T}, \\
\lambda_y(0) &= 1, \\
H|_{x=0} &= [-T + \lambda_y \dot{y} + \lambda_z \dot{z} + \lambda_r T] |_{x=0} = 0 \text{ (Hamiltonian)}, \\
H|_{x=1} &= 0.
\end{align*}
\]

Nodes: Initial data:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.</td>
<td>y = 1.</td>
<td>z = 3.1</td>
<td>\lambda_y = 1.</td>
</tr>
<tr>
<td>0.5</td>
<td>y = 15.</td>
<td>z = 6.</td>
<td>\lambda_y = 0.05</td>
</tr>
<tr>
<td>1.</td>
<td>y = 70.</td>
<td>z = 70.</td>
<td>\lambda_y = 0.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>29.0</td>
<td>19.0</td>
<td>11597</td>
<td>19</td>
</tr>
<tr>
<td>VOAS</td>
<td>42.4</td>
<td>32.9</td>
<td>4893</td>
<td>27</td>
</tr>
<tr>
<td>RKF7</td>
<td>94.9</td>
<td>84.8</td>
<td>14164</td>
<td>61</td>
</tr>
<tr>
<td>RKF4</td>
<td>460.6</td>
<td>410.9</td>
<td>74486</td>
<td>294</td>
</tr>
</tbody>
</table>

Example 14. Planar Earth–Mars transfer \((N = 9)\).

In interplanetary space science it is required to pilot an ion rocket with minimal mass-loss. In this example a travel from the Earthian orbit to the Martian orbit is treated. Both orbits are assumed to be circular and coplanar. The influence of the gravitational field of the Earth is neglected. This leads to an optimal control problem with two control variables: the thrust \(\beta \in [0, \beta_{\text{max}}]\) (linear control variable) and the thrust angle \(\psi \in ]-\pi, \pi[\) (nonlinear control variable).

For the (normed) physical quantities (distance from the sun \(r\), radial-velocity \(\omega\), tangential-velocity \(v\), mass \(m\)) the four differential equations hold:

\[
\begin{align*}
\dot{r} &= \omega, \\
\dot{\omega} &= \frac{v^2}{r} - \frac{1}{r^2} + \beta \frac{c}{m} \sin \psi; \quad c = 1.872, \\
\dot{v} &= -\frac{\omega v}{r} + \beta \frac{c}{m} \cos \psi; \quad \beta_{\text{max}} = 0.075, \\
\dot{m} &= -\beta.
\end{align*}
\]

The boundary values are given by:

\[
\begin{align*}
r(0) &= 1, \quad r(t_f) = 1.525, \\
\omega(0) &= 0, \quad \omega(t_f) = 0, \\
v(0) &= 1, \quad v(t_f) = 0.8098, \\
m(0) &= 1.
\end{align*}
\]

The problem with free total flight time \(t_f\) is yet unsolved.
Nodes: equidistant.
Initial data: solution at $t_i = 5.0$.

<table>
<thead>
<tr>
<th></th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSYS1</td>
<td>46.4</td>
<td>20.2</td>
<td>1220</td>
<td>53</td>
</tr>
<tr>
<td>VOAS</td>
<td>54.8</td>
<td>37.9</td>
<td>5870</td>
<td>63</td>
</tr>
<tr>
<td>RKF7</td>
<td>82.2</td>
<td>61.9</td>
<td>8967</td>
<td>94</td>
</tr>
<tr>
<td>RKF4</td>
<td>166.5</td>
<td>122.7</td>
<td>19470</td>
<td>190</td>
</tr>
</tbody>
</table>


Since the development of the Space Shuttle atmospheric entry of lifting vehicles is studied. These lifting vehicles should be capable of considerable lateral range, which allows increased return frequencies from orbit to given landing sites. For maximum lateral range a heating constraint depending on velocity, altitude, and angle of attack is taken into account.

The underlying mathematical model of this optimal control problem is due to Dickmanns [14], who also prepared initial data for the unconstrained problem (no heating constraint) using an analytical approximate solution. In [31] the numerical solution of the constrained problem has been treated using multiple shooting techniques especially trimmed to handle ill-conditioned two-point boundary value problems (cf. [11]). An extensive presentation of the whole model would be beyond the scope of this paper, it can be found in [13]. The heating constraint was pushed down from $2850\degree$F to $1700\degree$F by means of a continuation method. The special homotopy step reduces the level of the permitted skin temperature of the Space Shuttle from $2057\degree$F to $2000\degree$F. Using the modified continuation method [13] the system comprises 14 differential equations.

The differential equations for the physical quantities (velocity $v$, heading angle $\chi$, flight path angle $\gamma$, cross-range angle $\Lambda$, altitude $h$, down-range angle $\theta$) are:

\[
\begin{align*}
\dot{v} & = -(c_{Wm} + k \cdot c_A^n) \frac{F_{\theta}}{2m} e^{-\beta h} v^2 - g_0 \left( \frac{R}{R+h} \right)^2 \sin \gamma, \\
\dot{\chi} & = c_A \frac{F_{\theta}}{2m} e^{-\beta h} v \frac{\sin \mu}{\cos \gamma} - \frac{v}{R+h} \cos \gamma \cos \chi \tan \Lambda, \\
\dot{\gamma} & = c_A \frac{F_{\theta}}{2m} e^{-\beta h} v \cos \mu - \left( \frac{g_0}{v} \left( \frac{R}{R+h} \right)^2 - \frac{v}{R+h} \right) \cos \gamma, \\
\dot{\Lambda} & = \frac{v}{R+h} \cos \gamma \sin \chi, \\
\dot{h} & = v \sin \gamma, \\
\dot{\theta} & = \frac{v}{R+h} \cdot \frac{\cos \gamma \cos \chi}{\cos \Lambda}
\end{align*}
\]

where $\mu$ (aerodynamic bank angle) and $c_A$ (lift coefficient, constrained) are the two nonlinear control variables. The other quantities are constants. For a detailed discussion of the technical results see [15].
This example is one of the most complicated and most sensitive control problems to be solved so far by multiple shooting algorithm.

Nodes and initial data can be found in [31].

<table>
<thead>
<tr>
<th></th>
<th>TIME</th>
<th>OVHM</th>
<th>NFC</th>
<th>NCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY1</td>
<td>3850.6</td>
<td>795.9</td>
<td>346370</td>
<td>35</td>
</tr>
<tr>
<td>VOAS</td>
<td>3071.9</td>
<td>1672.7</td>
<td>146689</td>
<td>28</td>
</tr>
<tr>
<td>RKF7</td>
<td>7252.9</td>
<td>3970.7</td>
<td>384485</td>
<td>66</td>
</tr>
<tr>
<td>RKF4</td>
<td>FAIL</td>
<td></td>
<td></td>
<td>272</td>
</tr>
</tbody>
</table>

Remark. RKF4 did not succeed in 10800 sec (number of computed trajectories: NT=30, number of function evaluations: NCF = 58296). To make the calculation of NCT for RKF4 possible, the time was extrapolated to 30000.

5. Summary of Results

The 15 examples presented in this paper may be considered as representative two-point boundary value problems. They cover different degrees of difficulties. Some of the examples are rather simple and have been treated repeatedly in the literature (expl. 1, 3, 4, 6), others are extremely difficult (expl. 11, 13, 14, 15).

The criteria for judging the four integration routines are reliability, fastness and precision.

a) Reliability. As far as reliability is concerned, DIFSY1 obtained the best results. DIFSY1 is not sensitive to the choice of nodes (expl 3, 4). It also runs well even in sensitive problems described by a large norm of the iteration matrix (cf. Chapter 3). A disadvantage of DIFSY1 is the increased computing time, if DIFSY1 is used with zero initial data in the case of a homogeneous differential equation.

As can be seen from the examples (3, 4), VOAS was the most sensitive integrator. It does not allow large integration intervals, and requires the greatest number of nodes. Nevertheless, VOAS proved to be more favourable in applications in optimal control problems. The small integration steps of VOAS yielded a secure and fast determination of the switching points.

RKF7 and RKF4 are very reliable, except that the stepsize control of RKF7 did not avoid an overflow in the right hand side of example 11. Both of these integrators did not allow as large parameter values as DIFSY1 did in the examples 3 and 4.

b) Fastness. In order to compare the fastness of the four routines, the total amount of computing time is considered. The number of function calls, which has been measured in [28] and [30], is not alone representative for fastness. Computing time must be paid, not the number of function calls. Even for complicated problems with a large number of operations in the right hand side, the overheadtime of the routines can not be neglected (see Table 1).

As is seen in Table 1, the mean of the overheadtime for all problems (in per cent) requires a larger portion of time than the time spent in the right hand side. Only in example 15 TFC is dominant in all routines. It should be mentioned that VOAS
Table 1. Splitting of the total computing time (100%) averaged over all examples

<table>
<thead>
<tr>
<th></th>
<th>DIFSY 1</th>
<th>VOAS</th>
<th>RKF7</th>
<th>RKF4</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVHM</td>
<td>47%</td>
<td>75%</td>
<td>61%</td>
<td>60%</td>
</tr>
<tr>
<td>TFC</td>
<td>40%</td>
<td>18%</td>
<td>29%</td>
<td>30%</td>
</tr>
<tr>
<td>OVHB</td>
<td>13%</td>
<td>7%</td>
<td>10%</td>
<td>10%</td>
</tr>
</tbody>
</table>

OVHM = percentage for the overhead time of the routines; TFC = percentage for the time spent in the right hand side; OVHB = percentage for the overhead time of Boundsol

Table 2. Averaged normed computing time in per cent of the RKF 7-average

<table>
<thead>
<tr>
<th></th>
<th>Expl. 1–11</th>
<th>Expl. 12–15</th>
<th>Expl. 1–15</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY 1</td>
<td>95</td>
<td>44</td>
<td>81</td>
</tr>
<tr>
<td>VOAS</td>
<td>152</td>
<td>49</td>
<td>123</td>
</tr>
<tr>
<td>RKF7</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>RKF4</td>
<td>135</td>
<td>254</td>
<td>169</td>
</tr>
</tbody>
</table>

needs the largest, DIFSY 1 the smallest part of OVHM. The overhead time of BOUNDSOL, however, is negligible.

It is not possible to compare the four routines by adding the total computing times for each problem. The various difficulties and sizes of the problems caused computing times ranging between a few seconds (expl. 1) and several hours (expl. 15). The normed computing times NCT introduced in Chapter 4 are comparable quantities.

Table 2 includes the averaged NCT for each routine given in per cent of the RKF 7-average in each column.

This table illustrates the fastness of each routine for the examples 1–11 with technical precision (TOL = 1.0 x 10^-6), and for the sensitive optimal control problems 12–15 with higher precision (TOL = 1.0 x 10^-9). Column three in Table 2 contains the averaged values over all problems. As is seen in examples 1–11 DIFSY 1, and RKF7 run similarly fast, whereas in examples 12–15 DIFSY 1 and VOAS are equivalent. RKF4 is out of business. Having considered all problems, one would most likely prefer DIFSY 1. The above division of the examples into two groups, however, is unfavourable for RKF4, since RKF4 was the fastest routine in the examples 2, 7, 9.

c) Precision. In shooting methods, the precision is prescribed 'a priori'. All methods obtained the required precision.

Rule of Thumb. In order to decide which routine should be applied for a particular problem, the 15 examples are divided into four groups (according to their averaged total computing time).

Simple examples: 2, 7, 9
Less difficult examples: 1, 3, 4, 5, 6, 8
Difficult examples: 10, 12, 14
Very difficult examples: 11, 13, 15.
For these four groups the averaged normed computing times for each routine are compared in Table 3. Again they are given in per cent of the RKF7-average in each column.

Table 3. Averaged normed computing time for the classification in per cent of the RKF7-average

<table>
<thead>
<tr>
<th></th>
<th>Simple</th>
<th>Less difficult</th>
<th>Difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFSY 1</td>
<td>77</td>
<td>104</td>
<td>63</td>
<td>58</td>
</tr>
<tr>
<td>VOAS</td>
<td>121</td>
<td>165</td>
<td>89</td>
<td>81</td>
</tr>
<tr>
<td>RKF 7</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>RKF 4</td>
<td>65</td>
<td>163</td>
<td>158</td>
<td>374</td>
</tr>
</tbody>
</table>

For simple problems (column 1) RKF 4 is the fastest routine, whereas for less difficult problems (column 2) one would prefer RKF 7 or DIFSY 1. In difficult and very difficult problems DIFSY 1 and VOAS are the fastest routines. Among very difficult problems, where the part of computing time spent in the right hand side dominates (see expl. 15), VOAS is faster than DIFSY 1. This is caused by the larger percentage of overhead time of VOAS (see Table 1) (i.e. the smaller number of function calls). These statements are summed up in the Figure 1 which also includes the experience of the authors.

![Fig. 1. Behaviour of the integration routines](image)

The results of these investigations validate the suggestions in [16], that the four tested routines should be included in a program library. The choice of a suitable routine for a particular problem is made easier by the sketch.

Acknowledgements. The authors wish to thank Professor R. Bulirsch who stimulated and encouraged this work, they are indebted to Dr. D. Gerber for helpful comments.

References