Robust and Reliable Defect Control for Runge-Kutta Methods

by

Li Yan

A thesis submitted in conformity with the requirements
for the degree of Master of Science
Graduate Department of Computer Science
University of Toronto

Abstract

Robust and Reliable Defect Control for Runge-Kutta Methods

Li Yan

Master of Science

Graduate Department of Computer Science

University of Toronto

2007

Using defect error control for a class of continuous Runge-Kutta methods for solving nonstiff problems has been studied over the last two decades. Recently a class of such methods, with an associated defect that can be reliably controlled, has been proposed [5]. This approach is improved in this thesis by increasing the degree of the local interpolant by one and eliminating one of the terms contributing to the leading coefficient in the asymptotic expansion of the defect. We demonstrate this new improved robust and reliable defect approach on three continuous Runge-Kutta methods of order 5, 6, and 8. We also plot the shape of defect curves and compare them with previous work. Numerical results on the 25 test problems of DETEST at a wide range of accuracy requests are presented as well.

Acknowledgements

This research project is funded by Natural Sciences and Engineering Research Council of Canada (NSERC). I would like to express my special thanks to my supervisor, Professor Wayne Enright, for his patience, expert guidance, and passionate support during my MSc program. I would also like to give special thanks to Professor Christina C. Christara for her helpful advice and careful comments on my work. Finally, I would like to thank my mother for her support and unconditional love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Using defect error control for a class of continuous Runge-Kutta methods (CRKs) for solving non-stiff problems has been studied over the last two decades. The advantage of this approach, which indirectly controls the global error by bounding the magnitude of the defect on each step of integration, is that the associated global error bound is independent of the method. Although there are several papers that discuss the robustness and reliability of this approach, our initial experimental results revealed that for some non-stiff problems the estimated maximum defect across a step did not behave as reliably as expected. This motivated us to a further study of the widely used defect estimates and to the development of a more reliable estimate.

1.2 Background

1.2.1 Runge-Kutta Methods for ODE IVP

Consider the numerical solution of the initial value problem (IVP)

$$y' = f(t, y), \quad y(t_0) = y_0, \quad \text{over } [t_0, t_F], \quad \text{with } y \in \Re^m,$$

using an explicit continuous Runge-Kutta method (1.4) that has an underlying embedded discrete formula pair of orders $p-1$ and p. To derive such a method, one generally adds an interpolating scheme of appropriate order to a particular discrete Runge-Kutta formula pair. See [6] for more details and examples of this approach.

Let the underlying p^{th}-order, explicit, $\tilde{s}$-stage, discrete Runge-Kutta formula be defined by

$$y_i = y_{i-1} + h\sum_{j=1}^{\tilde{s}} w_j k_j, \tag{1.1}$$

where

$$k_j = f(t_{i-1} + c_j h, Y_j), \tag{1.2}$$

$$Y_j = y_{i-1} + h\sum_{r=1}^{j-1} a_{jr} f(t_{i-1} + c_r h, Y_r) \equiv y_{i-1} + h\sum_{r=1}^{j-1} a_{jr} k_r. \tag{1.3}$$

The values of w_j, c_j, and a_{jr} that define this formula are represented by $(w_1, w_2, \ldots, w_{\tilde{s}})$ and the first $\tilde{s}$ rows of the tableau shown in Table 1.1.

One can analyze the error in CRK methods by assuming that $u_i(t)$ defined for $t \in [t_{i-1}, t_i]$ is an approximation to the solution of the local IVP

$$u' = f(t, u), \quad u(t_{i-1}) = y_{i-1}, \quad \text{over } [t_{i-1}, t_i].$$

A standard (non-optimal) p^{th}-order interpolating polynomial $z_i(t)$ can be derived for this discrete formula (see [6] for a discussion of how this can be done and for some specific examples).

$$z_i(t) = y_{i-1} + h\sum_{j=1}^{s} b_j(\tau) k_j, \quad t \in [t_{i-1}, t_i], \tag{1.4}$$

where the k_j are determined by (1.2) and (1.3), and the $b_j(\tau)$ are polynomials,

$$b_j(\tau) = \sum_{k=1}^{p+1} \beta_{jk} \tau^k, \tag{1.5}$$

where

$$\tau = (t - t_{i-1})/h. \tag{1.6}$$

The interpolant associated with this step and defined by (1.4), (1.5) is represented by

$$\begin{array}{c|ccccc}
0 & 0 & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s,1} & a_{s,2} & \dots & a_{s,s-1} & \\
\hline
 & w_1 & w_2 & \dots & w_{s-1} & w_s \\
 & b_1(\tau) & b_2(\tau) & \dots & b_{s-1}(\tau) & b_s(\tau)
\end{array}$$

Table 1.1: The tableau for an explicit CRK method.

the tableau shown in Table 1.1.

A more accurate $(p+1)^{th}$-order interpolating polynomial $u_i(t)$ can be derived (again see [6] for details) which requires extra $(\bar{s}-s)$ stages. This interpolant can be represented by adding $(\bar{s}-s)$ rows to the tableau of Table 1.1 and introducing a new vector of polynomials $\bar{b}_j(\tau)$. Then

$$u_i(t) = y_{i-1} + h\sum_{j=1}^{\bar{s}} \bar{b}_j(\tau)k_j, \quad t \in [t_{i-1}, t_i], \tag{1.7}$$

where

$$\bar{b}_j(\tau) = \sum_{k=1}^{p+1} \bar{\beta}_{jk}\tau^k. \tag{1.8}$$

Several examples of the derivation and a discussion of their use in IVP software in formulas of orders four through eight are presented in Enright [4].

1.2.2 Defect

Numerical methods attempt to ensure that the accuracy of the continuous approximate solution is proportional to the specified tolerance. This is traditionally done by controlling the discrete local error on each step. However, the relationship between the local error and the global error is dependent on the method and problem. An alternative error control is based on monitoring and controlling an associated defect. This approach is based on forming an explicit interpolant on each attempted step and evaluating the corresponding defect at one or more fixed points before the decision to accept the step. The defect of an interpolant $u(t)$ is defined to be

$$\delta(t) = u'(t) - f(t, u(t)). \tag{1.9}$$

It has been shown [3] that for a $(p+1)^{st}$-order CRK with $u_i(t)$, defined by (1.7), the corresponding defect satisfies

$$\delta(t) = G(\tau)h^p + O(h^{p+1}), \tag{1.10}$$

where

$$G(\tau) = \sum_{j=1}^{m} q_j(\tau)F_j, \tag{1.11}$$

the F_j are problem-dependent, but independent on h, and the polynomials, $q_j(\tau)$, depend only on the method. If we can determine an interpolant $u_i(t)$ with $m = 1$ (or with $m = 2$ with the polynomial $q_1(\tau)$ dominant) in (1.11), the leading term in the expression of the defect, $G(\tau)$, will be dominated by $q_1(\tau)F_1$. Therefore, in the case that q_1 is dominant, $G(\tau)$ should approach a constant polynomial as $h \to 0$. Hence, the "shape" of defect curve should be dominated by the shape of the pre-determined polynomial $q_1(\tau)$, independent of the problem and the step. (Note that, since F_1 will in general depend on the problem and the step number, the corresponding $G(\tau)$ from (1.11) will be a constant multiple of $q_1(\tau)$ when $m = 1$. It is in this sense, we say that the $G(\tau)$ will have the same shape for all problems.) The location of the local maximum defect (as $h \to 0$) can then be pre-determined [3]. At run-time, we just need a single evaluation of the defect at this pre-determined point to obtain a reliable estimated maximum defect. We use the Lagrange form of the interpolating polynomials to define and analyze the $q_j(\tau)$ in the next section.

1.2.3 Lagrange Form

Given a set of data points $(t_0, y_0), \ldots, (t_n, y_n)$, with the t_j distinct $(t_0 < t_1 \ldots < t_n)$, there is a unique polynomial of degree at most n passing through these $n+1$ support points. This unique polynomial can be written in terms of the Lagrange basis

$$P_n(t) = y_0 L_0(t) + y_1 L_1(t) + \ldots + y_n L_n(t) = \sum_{j=0}^{n} y_j L_j(t), \tag{1.12}$$

where

$$L_j(t) = \prod_{k=0 \wedge k \neq j}^{n} \frac{t - t_k}{t_j - t_k}.$$

If t_k is a support point, then

$$L_j(t_k) = \begin{cases} 0 & \text{if } j \neq k \\ 1 & \text{if } j = k. \end{cases}$$

The $L_j(t)$ are called the n^{th}-degree Lagrange interpolating polynomials, associated with the points $(t_0, t_1, \ldots, t_n)$.

If the derivative values y'_j are known, we can form an associated polynomial that interpolates y_j and y'_j. We can use an extension of the Lagrange polynomials to write the Lagrange form of this interpolant P_{2n+1}. In the case we are interested in the number of derivative values y'_j and number of solution values y_j (to be interpolated) are not equal. More specifically, we will consider the case where two solution values and $(n+1)$

derivative values are specified. That is, we are given

$$
\begin{aligned}
&t_0, t_1, \ldots, t_n, \\
&y_i, \quad \text{for } i = 0, 1, \\
&y'_j, \quad \text{for } j = 0, \ldots, n.
\end{aligned}
$$

We can write the corresponding interpolant as the unique polynomial of degree at most $n+2$ which satisfies these $n+3$ interpolation conditions. Let this unique polynomial be represented by

$$P_{n+2}(t) = y_0 Q_0(t) + y_1 Q_1(t) + \sum_{j=0}^{n} Q_{2+j}(t) y'_j. \tag{1.13}$$

We then have

$$P'_{n+2}(t) = y_0 Q'_0(t) + y_1 Q'_1(t) + \sum_{j=0}^{n} Q'_{2+j}(t) y'_j, \tag{1.14}$$

where Q_j are the $(n+2)$-degree generalised Lagrange interpolating polynomials. That is, the polynomial $P_{n+2}(t)$ satisfies the $n+3$ constraints,

$$P_{n+2}(t_0) = y_0, \tag{1.15}$$

$$P_{n+2}(t_1) = y_1, \tag{1.16}$$

$$P'_{n+2}(t_j) = y'_j, \quad \text{for } j = 0, \ldots, n. \tag{1.17}$$

Since these equations must be satisfied for any value of $y_0, y_1, y'_0, y'_1, \ldots, y'_n$, we have, from (1.13) and (1.15)

$$
\begin{cases}
Q_0(t_0) = 1 \\
Q_1(t_0) = 0 \\
Q_{2+j}(t_0) = 0, \quad \text{for } j = 0, \ldots, n.
\end{cases}
\tag{1.18}
$$

(1.13) and (1.16) imply

$$\begin{cases} Q_0(t_1) = 0 \\ Q_1(t_1) = 1 \\ Q_{2+j}(t_1) = 0, \quad \text{for } j = 0, \ldots, n. \end{cases} \tag{1.19}$$

Relations (1.14) and (1.17) imply

$$\text{when } j = 0, \quad \begin{cases} Q_0'(t_0) = 0 \\ Q_1'(t_0) = 0 \\ Q_2'(t_0) = 1 \\ Q_{2+j}'(t_0) = 0, \quad \text{for } j = 1, \ldots, n; \end{cases} \tag{1.20}$$

$$\text{when } j = 1, \quad \begin{cases} Q_0'(t_1) = 0 \\ Q_1'(t_1) = 0 \\ Q_3'(t_1) = 1 \\ Q_{2+j}'(t_1) = 0, \quad \text{for } j = 0, 2, \ldots, n; \end{cases} \tag{1.21}$$

$$\vdots \tag{1.22}$$

$$\text{when } j = n, \quad \begin{cases} Q_0'(t_n) = 0 \\ Q_1'(t_n) = 0 \\ Q_{2+n}'(t_n) = 1 \\ Q_{2+n}'(t_n) = 0, \quad \text{for } j = 0, \ldots, n-1. \end{cases} \tag{1.23}$$

Note that (1.18), (1.19), ..., (1.23) represent $n+1$ sets of equations. Each set consists of $n+3$ linear equations, where the unknown are the $n+3$ sets of coefficients defining the polynomials $Q_0, Q_1, \ldots, Q_{n+2}$. These equations are trivial to decouple into $n+1$ sets of linear equations which are used to solve separately for $Q_0(t), Q_1(t), \ldots, Q_{n+2}(t)$.

Solving (1.18) – (1.23) gives us the unique polynomials $Q_0(t), Q_1(t), \ldots, Q_{n+2}(t)$, which from (1.13) determine the unique $P_{n+2}(t)$. In our investigation we will use this approach to develop and analyze properties of a local interpolant, $P(t)$, associated with the step from (t_{i-1}, y_{i-1}) to (t_i, y_i). Typically, we want $P(t)$ to interpolate some of the approximate solution values and derivative values associated with the i^{th} step. For example, the polynomial $P_{p+1}(t)$ of degree $\leq p+1$, that interpolates the four values $y_{i-1}, y_i, y'_{i-1}(\equiv k_1), y'_i$ as well as the additional $(p+2)$ derivative approximations,

$$k_r \approx y'(t_{i-1} + c_r h), \quad r = s-p+3, s-p+4, \ldots, s,$$

is, where written as a polynomial in $\tau = (t - t_{i-1})/h$,

$$P_{p+1}(\tau) = Q_0(\tau)y_{i-1} + Q_1(\tau)y_i + hQ_2(\tau)k_1 + hQ_3(\tau)y'_i + h\sum_{j=1}^{p-2} Q_{3+j}(\tau)k_{s+2-p+j}. \quad (1.24)$$

That is, $P_{p+1}(\tau)$ will interpolate the last $(p-2)$ stage values, $k_s, k_{s-1}, \ldots, k_{s-p+3}$. Note that writing $P_{p+1}(t)$ as a polynomial in τ has the advantage that the corresponding $Q_j(\tau)$ are independent of i, depending only on h and $(c_{s-p+3}, c_{s-p+2}, \ldots, c_s)$. On the other hand, the derivatives that appear in (1.20) - (1.23), $Q'_k(t)$, are derivatives with regard to t and we must use

$$\frac{dQ_k(\tau)}{dt} = \frac{1}{h}(\frac{dQ_k(\tau)}{d\tau}).$$

Let $q_j(\tau)$ be the derivative of $Q_j(\tau)$ (i.e., $q_j(\tau) = \frac{dQ_k(\tau)}{dt}$), $q_j(\tau) = Q'_j(\tau)$. From (1.24) we see that the $Q_1(\tau)$ directly relates the local error in y_i to the error in the approximation $P_{p+1}(t_{i-1} + \tau h)$.

We can easily solve for $Q_j(\tau)$ and $q_j(\tau)$ in Maple, as they satisfy

$$\begin{array}{llll}
Q_0(0) = 1, & Q_1(0) = 0, & Q_2(0) = 0, & Q_{1+j}(0) = 0, \text{ for } j = 2, \ldots, p-2, \\
Q_0(1) = 0, & Q_1(1) = 1, & Q_2(1) = 0, & Q_{1+j}(1) = 0, \text{ for } j = 2, \ldots, p-2, \\
q_0(0) = 0, & q_1(0) = 0, & q_2(0) = 1, & q_{1+j}(0) = 0, \text{ for } j = 2, \ldots, p-2, \\
q_0(c_{s-p+3}) = 0, & q_1(c_{s-p+3}) = 0, & q_3(c_{s-p+3}) = 1, & q_{1+j}(c_{s-p+3}) = 0, \text{ for } j = 1, \ldots, p-2 \wedge j \neq 2 \\
 & & \cdots & \\
q_0(c_s) = 0, & q_1(c_s) = 0, & q_{p-2}(c_s) = 1, & q_{1+j}(c_s) = 0, \text{ for } j = 1, \ldots, p-2 \wedge j \neq (p-3) \\
q_0(1) = 0, & q_1(1) = 0, & q_{p-1}(1) = 1, & q_{1+j}(1) = 0, \text{ for } j = 1, \ldots, p-3,
\end{array}$$

where we assume $0 < c_{s-p+3} < c_{s-p+2} < \ldots < c_s < 1$ and $\tau \in [0,1]$.

1.3 A Review of Previous Work

Recently, Enright and Hayes [5] have proposed a class of CRKs where, at a cost of a few additional f-evaluations per step, the shape of the associated defect will be almost independent of the problem as $h \to 0$. Then the defect of this improved interpolant $v_i(t)$ has relatively constant "shape", where the term "shape" means that two functions have the same shape if one is a constant multiple of the other. The advantage of this same "shape" property for the defect is that we can pre-compute the point in the step where the maximum defect is expected to occur, and then with a single evaluation at that point a reliable estimate of the maximum defect can be obtained at run-time.

One can analyze the error in CRK methods by assuming that $u_i(t)$ defined for $t \in [t_{i-1}, t_i]$ is an approximation to the solution of the local IVP

$$u' = f(t,u), \quad u(t_{i-1}) = y_{i-1}, \quad \text{over } [t_{i-1}, t_i].$$

Consider using an interpolant that interpolates $u(t)$ exactly at t_{i-1}, t_i, and $u'(t)$ exactly at $t_{i-1}, t_i, t_{i-1} + c_r h$, for $r = s-p+3, s-p+4, \ldots, s$. The resulting unique interpolant of degree $\leq p+1$, $\tilde{P}(t) \approx u(t)$ (on step i) has an associated interpolation error IE_i that satisfies

$$IE_i = \frac{u^{n+2}\eta}{(n+2)!} h^{n+2} \tau^2 (\tau-1)^2 \prod_{r=s-p+3}^{s} (\tau - c_r), \quad \text{for some } \eta \in [t_{i-1}, t_i].$$

Now since we do not know the local solution $u(t)$, we approximate the data to be interpolated in the definition of $\tilde{P}(t)$, i.e., the values $[y_{i-1}, y'_{i-1}, u(t_i), u'(t_i), u'(t_{i-1} + c_r h); r = s-p+3, s-p+4, \ldots, s]$ are approximated by $[y_{i-1}, y'_{i-1}, y_i, y'_i, k_r; r = s-p+3, s-p+4, \ldots, s]$ and it is these values that define our approximating $P(t) \approx u(t)$ that is computed on step i (see (1.24)). The total error in this approximation can be written as

$$\begin{aligned} u(t) - P(t) &= [u(t) - \tilde{P}(t)] + [\tilde{P}(t) - P(t)] \\ &= IE_i + DE_i \end{aligned}$$

In [5] it was assumed that $n = p-1$ and that IE_i (which is $O(h^{p+1})$) is negligible relative to the data error. Note that by writing $\tilde{P}(t)$ and $P(t)$ in this Lagrange extension form, we observe (from (1.24))

$$\tilde{P}(t) = Q_0(\tau)y_{i-1} + Q_1(\tau)u(t_i) + hQ_2(\tau)k_1 + hQ_3(\tau)u'(t_i) + h\sum_{r=1}^{p-2} Q_{3+r}(\tau)u'(t_{i-1} + c_{s-p+2+r}h),$$

$$P(t) = Q_0(\tau)y_{i-1} + Q_1(\tau)y_i + hQ_2(\tau)k_1 + hQ_3(\tau)y_i' + h\sum_{r=1}^{p-2} Q_{3+r}(\tau)k_{s-p+2+r},$$

and therefore since the terms involving $Q_0(\tau)$ and $Q_2(\tau)$ are equal, subtracting these two equations, we obtain

$$\begin{aligned} DE_i &= \tilde{P}(t) - P(t) \\ &= Q_1(\tau)(u(t_i) - y_i) + hQ_3(\tau)(u'(t_i) - y_i') + h\sum_{r=1}^{p-2} Q_{3+r}(\tau)(u'(t_{i-1} + c_{s-p+2+r}h) - k_r). \end{aligned}$$

For a p^{th}-order CRK, we know $u(t_i) - y_i = O(h^{p+1})$ and we will usually observe that the data error will dominate the interpolation error as $h \to 0$. In our preliminary testing of the methods of this type identified in [5] we observed that for some problems, on some steps, the defect did not exhibit the expected shape as $h \to 0$. We observed that on these steps the interpolation error was not insignificant relative to the data error. When we introduced an additional interpolation stage and increased the degree of $\tilde{P}(t)$ and $P(t)$, we obtained better performance as the corresponding $IE_i = O(h^{p+2})$ while the data error remained $O(h^{p+1})$. This had the effect of forcing $m = 1$ in (1.11) and ensuring that the shape of $q_1(\tau)$ should reflect the shape of the defect on all steps and all problems as $h \to 0$.

1.4 Contributions of the Thesis

The main contribution of this thesis is that a more robust and reliable defect control for a class of continuous Runge-Kutta methods, specially for 5^{th}, 6^{th}, and 8^{th} order, is developed. We have done extensive experiments on several problems, and found that the interpolation error may not be negligible for some cases. So the interpolation error may cause an inconsistent shape of the defect curves across a step. Our new approach is to force the assumption in [5] to be true, at least, asymptotically, as $h \to 0$, by increasing the degree of the interpolant by one. For instance, for the CRK 4/5, we use a 6^{th}-degree interpolating polynomial; for the 5/6 CRK, we use a 7^{th}-degree interpolating polynomial, and for the 7/8 CRK, we use a 9^{th}-degree interpolating polynomial.

1.5 The Outline of the Thesis

In Chapter 2, we introduce a new improved 5^{th}-order CRK method, including the standard 5^{th}-order interpolant, and the derivation of the new improved interpolant and its associated $q_1(\tau)$. We also discuss the true defect of a simple example $y' = y$, and the improvement of the defect curves of some interesting problems compared with those of [5]. At the end of Chapter 2, the numerical results on the 25 test problems of DETEST [7] at a wide range of accuracy are presented. Chapter 3 and Chapter 4 introduce a 6^{th} and an 8^{th} order continuous Runge-Kutta method respectively. Both chapters have the similar structure with Chapter 2. In Chapter 5, we summarize the thesis and discuss future work.

Chapter 2

The 4/5 Pair

2.1 Standard 5^{th}-order Interpolating Polynomial

The tableau for a widely used 5^{th}-order, explicit, 7-stage Runge-Kutta 4/5 pair is shown in Table 2.1. This is the formula used in ODE45 [1] of Matlab.

$$
\begin{array}{c|ccccccc}
0 & 0 & & & & & & \\
\frac{1}{5} & \frac{1}{5} & 0 & & & & & \\
\frac{3}{10} & \frac{3}{40} & \frac{9}{40} & 0 & & & & \\
\frac{4}{5} & \frac{44}{45} & -\frac{56}{15} & \frac{32}{9} & 0 & & & \\
\frac{8}{9} & \frac{19372}{6561} & -\frac{25360}{2187} & \frac{64448}{6561} & -\frac{212}{729} & 0 & & \\
1 & \frac{9017}{3168} & -\frac{355}{33} & \frac{46732}{5247} & \frac{49}{176} & -\frac{5103}{188656} & 0 & \\
1 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & 0 \\
\hline
 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & 0
\end{array}
$$

Table 2.1: The tableau of the CRK45.

The 5^{th}-order discrete solution y_i is defined by

$$y_i = y_{i-1} + h \sum_{j=1}^{6} w_j k_j, \tag{2.1}$$

where

$$k_j = f(t_{i-1} + c_j h, Y_j), \tag{2.2}$$

$$Y_j = y_{i-1} + h \sum_{r=1}^{j-1} a_{jr} f(t_{i-1} + c_r h, Y_r). \tag{2.3}$$

Note one of the extra stages (in this case k_7) that is used in the definition of $z_i(t)$ is equal to $y_i' = f(t_{i-1} + h, y_i)$. The standard (non-optimal) 4^{th}-order interpolating polynomial $z_i(t)$ that agrees with the local solution to $(O(h^5))$ is defined by

$$z_i(t) = y_{i-1} + h \sum_{j=1}^{7} b_j(\tau) k_j, \quad \tau \in [t_{i-1}, t_i], \tag{2.4}$$

where

$$\begin{bmatrix} b_1(\tau) \\ b_2(\tau) \\ b_3(\tau) \\ b_4(\tau) \\ b_5(\tau) \\ b_6(\tau) \\ b_7(\tau) \end{bmatrix} = \begin{bmatrix} 1 & -\frac{183}{64} & \frac{37}{12} & -\frac{145}{128} \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1500}{371} & -\frac{1000}{159} & \frac{1000}{371} \\ 0 & -\frac{125}{32} & \frac{125}{12} & -\frac{375}{64} \\ 0 & \frac{9477}{3392} & -\frac{729}{106} & \frac{25515}{6784} \\ 0 & -\frac{11}{7} & \frac{11}{3} & -\frac{55}{28} \\ 0 & \frac{3}{2} & -4 & \frac{5}{2} \end{bmatrix} \begin{bmatrix} \tau \\ \tau^2 \\ \tau^3 \\ \tau^4 \end{bmatrix}.$$

Note that the defect associated with $z_i(t)$ will be $O(h^4)$ for $t \in [t_{i-1}, t_i]$. To construct 5^{th}-order interpolant $u_i(t)$, two additional stages, k_8 and k_9, are introduced (see [2, 3] for a discussion of this particular choice),

$$k_8 = f(t_{i-1} + .86h, z_i(t_{i-1} + .86h)), \tag{2.5}$$

$$k_9 = f(t_{i-1} + .93h, z_i(t_{i-1} + .93h)), \tag{2.6}$$

and $u_i(t)$ is defined using these 9 stages as

$$u_i(t_{i-1} + \tau h) = y_{i-1} + h \sum_{j=1}^{9} \bar{b}_j(\tau) k_j, \tag{2.7}$$

where

$$\begin{bmatrix} \bar{b}_1(\tau) \\ \bar{b}_2(\tau) \\ \bar{b}_3(\tau) \\ \bar{b}_4(\tau) \\ \bar{b}_5(\tau) \\ \bar{b}_6(\tau) \\ \bar{b}_7(\tau) \\ \bar{b}_8(\tau) \\ \bar{b}_9(\tau) \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1708582621}{524156928} & \frac{1232939669}{262078464} & -\frac{1663764925}{524156928} & \frac{208375}{253952} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{499875}{94976} & -\frac{1618625}{142464} & \frac{871875}{94976} & -\frac{15625}{5936} \\ 0 & \frac{499875}{65536} & -\frac{1618625}{98304} & \frac{871875}{65536} & -\frac{15625}{4096} \\ 0 & -\frac{26237439}{6946816} & \frac{28319463}{3473408} & -\frac{45762975}{6946816} & \frac{820125}{434176} \\ 0 & \frac{43989}{28672} & -\frac{142439}{43008} & \frac{76725}{28672} & -\frac{1375}{1792} \\ 0 & -\frac{2291427}{100352} & \frac{3838251}{50176} & -\frac{8579075}{100352} & \frac{199625}{6272} \\ 0 & -\frac{47953125}{1078784} & \frac{74828125}{539392} & -\frac{155453125}{1078784} & \frac{78125}{1568} \\ 0 & \frac{8734375}{145824} & -\frac{14359375}{72912} & \frac{31234375}{145824} & -\frac{234375}{3038} \end{bmatrix} \begin{bmatrix} \tau \\ \tau^2 \\ \tau^3 \\ \tau^4 \\ \tau^5 \end{bmatrix}.$$

Note that for $t \in [t_{i-1}, t_i]$, $u_i(t)$ will agree with the local solution $u(t)$ to $O(h^6)$ and the associated defect will be $O(h^5)$.

2.2 Derivative of $q_1(t)$ for the 4/5

As we mentioned in Chapter 1.2.3, the $q_j(\tau)$ arising in the definition of $G(\tau)$ are the derivatives of the Q_j which are the generated Lagrange interpolating polynomials associated with the p data points $(0, c_{s-p+3}, c_{s-p+4}, \ldots, c_s, 1)$. We call $[c_{s-p+3}, c_{s-p+4}, \ldots, c_s]$ the abscissa vector.

In [5], a degree 5 interpolating polynomial is derived using the above $u_i(t)$ (2.7) to define two additional, more accurate stages k_{10} and k_{11}. The assumption was that the data errors would dominate the interpolation error. The abscissa vector associated with these two new stages is $[0.1, 0.9]$. For our new improved continuous Runge-Kutta 4/5 method, called CRK45, we chose a 6^{th}-degree Lagrange polynomial agreeing with the local solution to $O(h^5)$ but with a associated interpolation error, that is $O(h^6)$. Let the abscissa used to generate these new stages, k_{10}, k_{11}, and k_{12}, be $[c_{10}, c_{11}, c_{12}]$. We can investigate properties of the resulting new interpolant using the approach of [5]. The following is an overview of the Maple program that given values for c_{10}, c_{12}, and c_{13} will compute the corresponding CRK.

$$Q_1(x) := b_{10} + b_{11}x + b_{12}x^2 + b_{13}x^3 + b_{14}x^4 + b_{15}x^5 + b_{16}x^6;$$

$$q_1(x) := b_{11} + 2b_{12}x + 3b_{13}x^2 + 4b_{14}x^3 + 5b_{15}x^4 + 6b_{16}x^5;$$

$$\text{solve}(\{Q_1(0) = 0, Q_1(1) = 1, q_1(0) = 0, q_1(1) = 0, q_1(c_{10}) = 0, q_1(c_{11}) = 0, q_1(c_{12}) = 0\},$$
$$\{b_{10}, b_{11}, b_{12}, b_{13}, b_{14}, b_{15}, b_{16}\});$$

...

$$Q_6(x) := b_{60} + b_{61}x + b_{62}x^2 + b_{63}x^3 + b_{64}x^4 + b_{65}x^5 + b_{66}x^6;$$

$$q_6(x) := b_{61} + 2b_{62}x + 3b_{63}x^2 + 4b_{64}x^3 + 5b_{65}x^4 + 6b_{66}x^5;$$

$$\text{solve}(\{Q_6(0) = 0, Q_6(1) = 0, q_6(0) = 0, q_6(1) = 0, q_6(c_{10}) = 0, q_6(c_{11}) = 0, q_6(c_{12}) = 1\},$$
$$\{b_{60}, b_{61}, b_{62}, b_{63}, b_{64}, b_{65}, b_{66}\});$$

The Maple source code for this step is included in Appendix B.1, and also given in `poly45exp.txt` which can be downloaded from our website. Solving these equations in Maple, we obtain Q_j and q_j for $j = 1, 2, \ldots, 6$. In choosing the most appropriate values for $[c_{10}, c_{11}, c_{12}]$, we look at various measures that have been used for quantifying the 'quality' of the corresponding interpolant. Such measures have been introduced and discussed in [4, 5, 6]. In this study we have used the following criteria,

- The maximum magnitude of $q_1(x)$ should not be large for $x \in [0, 1]$. In particular it is expected to be greater than 2 but less than 2.5.

- The ratio of the maximum magnitude of $q_j(x)$ (for $j \neq 1$) and $q_1(x)$ should be as small as possible to ensure that $q_1(x)$ dominates the other $q_j(x)$. Note that each of the other $q_j(x)$ will contribute to the $O(h^{p+1})$ term of (1.10).

Setting c_{10}, c_{11}, c_{12} close to either of the two end points, 0 or 1, makes it easier to satisfy the above criteria since (as is discussed in [5]) $q_1(x)$ changes sign at $[0, c_{10}, c_{11}, c_{12}, 1]$ and

$\int_0^1 q_1(\tau)d(\tau) = 1$. Figures 2.1 and 2.2 show the plot of q_j of some experiments. Figure 2.3 shows the plot of q_j which 'best' satisfies the above criteria. In this case, c_1, c_2, and c_3 are 0.10, 0.80, 0.90 respectively. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.57. The maximum of q_1 occurs at $\tau^* = 0.39$, $\tau \in [0, 1]$. The plot of q_1 and Q_1 for this formula is shown in Figure 2.4.

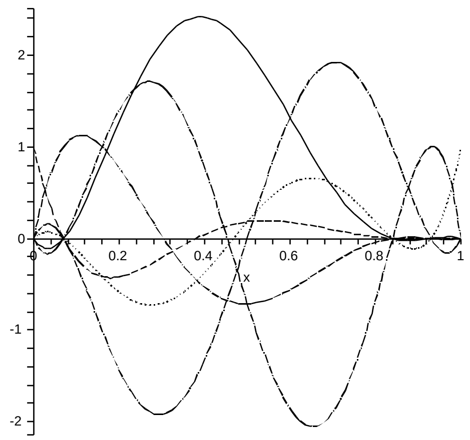

Figure 2.1: The plot of $q_1 \sim q_5$. The q_1 is presented by the solid line and has the highest magnitude among all q_j. The abscissa vector is [0.07, 0.84, 0.93]. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.79.

2.3 Formula of the New 4/5

To eliminate the contribution of the interpolation error to the coefficient of h^5 in the expression of the defect, we introduced one more point in the abscissa vector to construct a 6^{th}-degree interpolating polynomial. The corresponding abscissa vector is $[c_{10}, c_{11}, c_{12}] = [0.1, 0.8, 0.9]$ that we chose and justified in the previous section 2.2. We compute k_{10}, k_{11},

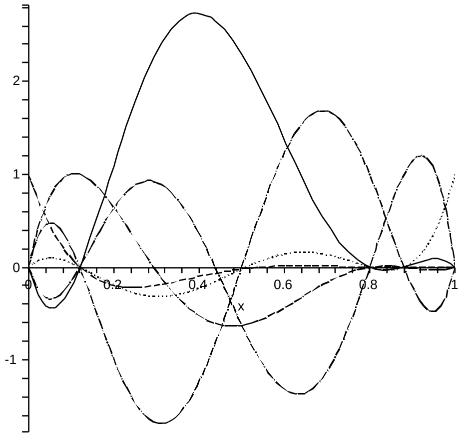

Figure 2.2: The plot of $q_1 \sim q_5$. The q_1 is presented by the solid line and has the highest magnitude among all q_j. The abscissa vector is [0.12, 0.80, 0.88]. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.62.

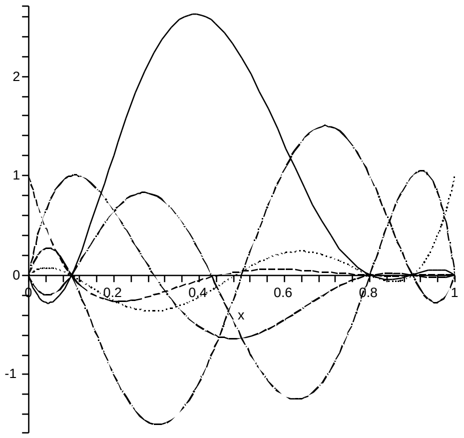

Figure 2.3: The plot of $q_1 \sim q_5$. The q_1 is presented by the solid line and has the highest magnitude among all q_j. The abscissa vector is [0.10, 0.80, 0.90]. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.57. The maximum of q_1 occurs at $\tau^* = 0.39$, $\tau \in [0, 1]$.

and k_{12} based on this new abscissa vector:

$$k_{10} = f(t_{i-1} + 0.1h, u_i(t_{i-1} + 0.1h)), \tag{2.8}$$

$$k_{11} = f(t_{i-1} + 0.8h, u_i(t_{i-1} + 0.8h)), \tag{2.9}$$

$$k_{12} = f(t_{i-1} + 0.9h, u_i(t_{i-1} + 0.9h)). \tag{2.10}$$

The new degree 6 interpolating polynomial can then be written as

$$v_i(t_{i-1} + \tau h) = Q_0(\tau)y_{i-1} + Q_1(\tau)y_i + Q_2(\tau)hy'_{i-1} + Q_3(\tau)hy'_i + h\sum_{j=1}^{3} Q_{3+j}(\tau)k_{9+j}. \tag{2.11}$$

Since we know (see [5] for details)

$$Q_0(\tau) = 1 - Q_1(\tau), \tag{2.12}$$

$$y_i = y_{i-1} + h\sum_{j=1}^{6} w_j k_j, \tag{2.13}$$

$$y'_{i-1} = k_1, \tag{2.14}$$

$$y'_i = k_7, \tag{2.15}$$

we can substitute (2.12), (2.13), (2.14), and (2.15) into (2.11), to obtain,

$$\begin{aligned} v_i(t_{i-1} + \tau h) &= (1 - Q_1(\tau))y_{i-1} + Q_1(\tau)(y_{i-1} + h\sum_{j=1}^{6} w_j k_j) + Q_2(\tau)hk_1 + Q_3(\tau)hk_7 \\ &\quad + h\sum_{j=1}^{3} Q_{3+j}(\tau)k_{9+j} \\ &= y_{i-1} + h\sum_{j=1}^{6} w_j Q_1(\tau)k_j + hQ_2(\tau)k_1 + hQ_3(\tau)k_7 + h\sum_{j=1}^{3} Q_{3+j}(\tau)k_{9+j} \\ &= y_{i-1} + h(w_1Q_1(\tau) + Q_2(\tau))k_1 + h\sum_{j=2}^{6} w_j Q_1(\tau)k_j + hQ_3(\tau)k_7 \\ &\quad + h\sum_{j=1}^{3} Q_{3+j}(\tau)k_{9+j}, \end{aligned}$$

which can be re-written as

$$v_i(t_{i-1} + \tau h) = y_{i-1} + h\sum_{j=1}^{12} \hat{b}_j(\tau)k_j, \tag{2.16}$$

where

$$\begin{cases} \hat{b}_1 = w_1 Q_1(\tau) + Q_2(\tau) \\ \hat{b}_2 = w_2 Q_1(\tau) \\ \hat{b}_3 = w_3 Q_1(\tau) \\ \hat{b}_4 = w_4 Q_1(\tau) \\ \hat{b}_5 = w_5 Q_1(\tau) \\ \hat{b}_6 = w_6 Q_1(\tau) \\ \hat{b}_7 = Q_3(\tau) \\ \hat{b}_8 = 0 \\ \hat{b}_9 = 0 \\ \hat{b}_{10} = Q_4(\tau) \\ \hat{b}_{11} = Q_5(\tau) \\ \hat{b}_{12} = Q_6(\tau). \end{cases}$$

The $Q_1 \sim Q_6$ can be easily solved in Maple. See Appendix B.1 for Maple source code. The corresponding coefficients defining the polynomial, $\hat{b}_j(\tau)$, are:

$$\begin{bmatrix} \hat{b}_1(\tau) \\ \hat{b}_2(\tau) \\ \hat{b}_3(\tau) \\ \hat{b}_4(\tau) \\ \hat{b}_5(\tau) \\ \hat{b}_6(\tau) \\ \hat{b}_7(\tau) \\ \hat{b}_8(\tau) \\ \hat{b}_9(\tau) \\ \hat{b}_{10}(\tau) \\ \hat{b}_{11}(\tau) \\ \hat{b}_{12}(\tau) \end{bmatrix} = \begin{bmatrix} 1 & -\frac{13303}{1584} & \frac{791347}{28512} & -\frac{1589515}{38016} & \frac{35045}{1188} & -\frac{113375}{14256} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12000}{4081} & \frac{962000}{36729} & -\frac{6725000}{12243} & \frac{80000}{1749} & -\frac{500000}{36729} \\ 0 & -\frac{375}{88} & \frac{60125}{1584} & -\frac{168125}{2112} & \frac{4375}{66} & -\frac{15625}{792} \\ 0 & \frac{19683}{9328} & -\frac{350649}{18656} & \frac{2941515}{74624} & -\frac{76545}{2332} & \frac{91125}{9328} \\ 0 & -\frac{6}{7} & \frac{481}{63} & -\frac{1345}{84} & \frac{40}{3} & -\frac{250}{63} \\ 0 & \frac{62}{33} & -\frac{16099}{891} & \frac{14095}{297} & -\frac{14620}{297} & \frac{16000}{891} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{2500}{231} & -\frac{304250}{6237} & \frac{170750}{2079} & -\frac{127250}{2079} & \frac{106250}{6237} \\ 0 & \frac{375}{56} & -\frac{15875}{252} & \frac{26125}{168} & -\frac{3125}{21} & \frac{3125}{63} \\ 0 & -\frac{500}{99} & \frac{43750}{891} & -\frac{39250}{297} & \frac{40750}{297} & -\frac{43750}{891} \end{bmatrix} \begin{bmatrix} \tau \\ \tau^2 \\ \tau^3 \\ \tau^4 \\ \tau^5 \\ \tau^6 \end{bmatrix}.$$

2.4 True Defect of the Example $y' = y$

To better understand the defect curve and verify the above analysis, we determine and plot the exact defect for the scalar problem $y' = y$ and compare it with $q_1(\tau)$.

Consider the simple scalar IVP

$$y' = y, \quad y(0) = 1,$$

with true solution $y(t) = e^t$. The detailed calculation of the true defect of this example for the first step is shown below and has been determined using Maple.

Using (2.2) and (2.3), we compute $k_1 \sim k_7$ for the first step,

$$
\begin{aligned}
k_1 &= 1,\\
k_2 &= 1 + \frac{1}{5}h,\\
k_3 &= 1 + \frac{9}{200}h^2 + \frac{3}{10}h,\\
k_4 &= 1 + \frac{4}{25}h^3 + \frac{8}{25}h^2 + \frac{4}{5}h,\\
k_5 &= 1 - \frac{848}{18225}h^4 + \frac{424}{1215}h^3 + \frac{32}{81}h^2 + \frac{8}{9}h,\\
k_6 &= 1 + \frac{7}{550}h^5 - \frac{14}{275}h^4 + \frac{21}{55}h^3 + \frac{1}{2}h^2 + h,\\
k_7 &= 1 + \frac{1}{600}h^6 + \frac{1}{120}h^5 + \frac{1}{24}h^4 + \frac{1}{6}h^3 + \frac{1}{2}h^2 + h,
\end{aligned}
$$

then we have

$$
y_1 = k_7 = 1 + \frac{1}{600}h^6 + \frac{1}{120}h^5 + \frac{1}{24}h^4 + \frac{1}{6}h^3 + \frac{1}{2}h^2 + h,
$$

and the standard interpolating polynomial (2.4) then is

$$
\begin{aligned}
z_1(\tau) = 1 &+ (\frac{1}{400}\tau^2 - \frac{1}{150}\tau^3 + \frac{1}{240}\tau^4)h^7 + (-\frac{3}{400}\tau^2 + \frac{1}{75}\tau^3 - \frac{1}{240}\tau^4)h^6\\
&+ (\frac{1}{80}h^2 - \frac{1}{30}h^3 + \frac{7}{240}h^4)h^5 + \frac{1}{24}\tau^4h^4 + \frac{1}{6}\tau^3h^3 + \frac{1}{2}\tau^2h^2 + \tau h.
\end{aligned}
$$

If we substitute $\tau = 1$ into $z_1(\tau)$, we can verify that

$$
\begin{aligned}
z_1(1) &= 1 + \frac{1}{600}h^6 + \frac{1}{120}h^5 + \frac{1}{24}h^4 + \frac{1}{6}h^3 + \frac{1}{2}h^2 + h\\
&= y_1.
\end{aligned}
$$

Then we compute k_8 and k_9 using (2.5) and (2.6),

$$
\begin{aligned}
k_8 = 1 &- \frac{168259}{1500000000}h^7 + \frac{327273}{500000000}h^6 + \frac{1998769}{500000000}h^5 + \frac{3418801}{150000000}h^4\\
&+ \frac{79507}{750000}h^3 + \frac{1849}{5000}h^2 + \frac{43}{50}h,\\
k_9 = 1 &- \frac{665973}{8000000000}h^7 + \frac{8969013}{8000000000}h^6 + \frac{46540269}{8000000000}h^5 + \frac{24935067}{800000000}h^4\\
&+ \frac{268119}{2000000}h^3 + \frac{8649}{20000}h^2 + \frac{93}{100}h,
\end{aligned}
$$

and determine $u_1(\tau)$ from (2.7)

$$\begin{aligned}u_1(\tau) &= 1 + (\frac{1}{1200}\tau^3 - \frac{1}{600}\tau^4 + \frac{1}{1200}\tau^5)h^8 + (-\frac{1}{400}\tau^3 + \frac{1}{300}\tau^4 - \frac{1}{1200}\tau^5)h^7 \\ &+ (\frac{1}{240}\tau^3 - \frac{1}{120}\tau^4 + \frac{7}{1200}\tau^5)h^6 + \frac{1}{120}\tau^5 h^5 + \frac{1}{24}\tau^4 h^4 + \frac{1}{6}\tau^3 h^3 + \frac{1}{2}\tau^2 h^2 + \tau h.\end{aligned}$$

We compute k_{10}, k_{11}, and k_{12} corresponding to $\tau = 0.1, 0.8, 0.9$ using (2.8), (2.9), and (2.10),

$$\begin{aligned}k_{10} &= 1 + \frac{27}{40000000}h^8 - \frac{87}{40000000}h^7 + \frac{407}{120000000}h^6 + \frac{1}{12000000}h^5 + \frac{1}{240000}h^4 \\ &+ \frac{1}{6000}h^3 + \frac{1}{200}h^2 + \frac{1}{10}h, \\ k_{11} &= 1 + \frac{4}{234375}h^8 - \frac{44}{234375}h^7 + \frac{148}{234375}h^6 + \frac{128}{46875}h^5 + \frac{32}{1875}h^4 + \frac{32}{375}h^3 \\ &+ \frac{8}{25}h^2 + \frac{4}{5}h, \\ k_{12} &= 1 + \frac{234}{40000000}h^8 - \frac{5103}{40000000}h^7 + \frac{40581}{40000000}h^6 + \frac{19683}{4000000}h^5 + \frac{2187}{80000}h^4 \\ &+ \frac{243}{2000}h^3 + \frac{81}{200}h^2 + \frac{9}{10}h.\end{aligned}$$

Finally, using (2.16), the improved new interpolation polynomial $v_1(\tau)$ for the first step is

$$\begin{aligned}v_1(\tau) &= 1 + (\frac{1}{11000}\tau^2 - \frac{481}{594000}\tau^3 + \frac{151}{79200}\tau^4 - \frac{173}{99000}\tau^5 + \frac{133}{237600}\tau^6)h^9 \\ &+ (-\frac{7}{11000}\tau^2 + \frac{3367}{594000}\tau^3 - \frac{991}{79200}\tau^4 + \frac{523}{49500}\tau^5 - \frac{733}{237600}\tau^6)h^8 \\ &+ (\frac{1}{440}\tau^2 - \frac{481}{23760}\tau^3 + \frac{689}{15840}\tau^4 - \frac{733}{19800}\tau^5 + \frac{2731}{237600}\tau^6)h^7 \\ &+ (-\frac{1}{550}\tau^2 + \frac{481}{29700}\tau^3 - \frac{269}{7920}\tau^4 + \frac{14}{495}\tau^5 - \frac{167}{23760}\tau^6)h^6 \\ &+ \frac{1}{120}\tau^5 h^5 + \frac{1}{24}\tau^4 h^4 + \frac{1}{6}\tau^3 h^3 + \frac{1}{2}\tau^2 h^2 + \tau h.\end{aligned}$$

The derivative of $v_1(\tau)$ is then

$$
\begin{aligned}
v_1'(\tau) = 1 &+ (\frac{1}{5500}\tau - \frac{481}{198000}\tau^2 + \frac{151}{19800}\tau^3 - \frac{173}{19800}\tau^4 + \frac{133}{39600}\tau^5)h^8 \\
&+ (-\frac{7}{5500}\tau + \frac{3367}{198000}\tau^2 - \frac{991}{19800}\tau^3 + \frac{523}{9900}\tau^4 - \frac{733}{39600}\tau^5)h^7 \\
&+ (\frac{1}{220}\tau - \frac{481}{7920}\tau^2 + \frac{689}{3960}\tau^3 - \frac{733}{3960}\tau^4 + \frac{2731}{39600}\tau^5)h^6 \\
&+ (-\frac{1}{275}\tau + \frac{481}{9900}\tau^2 - \frac{269}{1980}\tau^3 + \frac{14}{99}\tau^4 - \frac{167}{3960}\tau^5)h^5 \\
&+ \frac{1}{24}\tau^4h^4 + \frac{1}{6}\tau^3h^3 + \frac{1}{2}\tau^2h^2 + \tau h.
\end{aligned}
$$

Therefore, the defect for this first step is $\delta_1(\tau) = v_1'(\tau) - f(t, v_1)$,i.e.,

$$
\begin{aligned}
\delta_1(\tau) = &(-\frac{1}{11000}\tau^2 + \frac{481}{594000}\tau^3 - \frac{151}{79200}\tau^4 + \frac{173}{99000}\tau^5 - \frac{133}{237600}\tau^6)h^9 \\
&+ (\frac{1}{5500}\tau - \frac{71}{39600}\tau^2 + \frac{1163}{594000}\tau^3 + \frac{299}{79200}\tau^4 - \frac{1427}{198000}\tau^5 + \frac{733}{237600}\tau^6)h^8 \\
&+ (-\frac{7}{5500}\tau + \frac{2917}{198000}\tau^2 - \frac{3541}{118800}\tau^3 + \frac{739}{79200}\tau^4 + \frac{733}{39600}\tau^5 - \frac{2731}{237600}\tau^6)h^7 \\
&+ (\frac{1}{220}\tau - \frac{2333}{39600}\tau^2 + \frac{9373}{59400}\tau^3 - \frac{133}{880}\tau^4 + \frac{179}{4400}\tau^5 + \frac{167}{23760}\tau^6)h^6 \\
&+ (-\frac{1}{275}\tau + \frac{481}{9900}\tau^2 - \frac{269}{1980}\tau^3 + \frac{14}{99}\tau^4 - \frac{5}{99}\tau^5)h^5,
\end{aligned}
$$

and $q_1(\tau)$ which agrees with $Q_1'(\tau)$ as determined from Appendix B.1 is

$$
q_1(\tau) = -\frac{1}{275}\tau + \frac{481}{9900}\tau^2 - \frac{269}{1980}\tau^3 + \frac{14}{99}\tau^4 - \frac{5}{99}\tau^5.
$$

A typical first step size that is chosen when solving this problem with $tol = 10^{-6}$ is $h = 0.15773933612005$. For this choice we have

$$
\begin{aligned}
\delta_1(\tau) = &0.8149008326e - 7\tau^6 - 0.4263141538e - 5\tau^5 + 0.1150583935e - 4\tau^4 \\
&- 0.1090836097e - 4\tau^3 + 0.3872290335e - 5\tau^2 - 0.2881172523e - 6\tau.
\end{aligned}
$$

We plot the exact defect $\delta_1(\tau)$ and the $q_1(\tau)$ in Figures 2.5 and 2.7 respectively. They illustrate that $\delta_1(\tau)$ and $q_1(\tau)$ have a consistent shape. The roots of $q_1(\tau)$ are $0, 0.1, 0.8, 0.9, 1$, and the roots of $\delta_1(\tau)$ are $0, 0.9945204839e{-}1, 0.7977279534, 0.8999856390, 1.000000111$, 49.51768493. Obviously, the roots of $\delta_1(\tau)$ for $h = 0.15773933612005$ and $q_1(\tau)$ are close.

In addition, another typical step size is chosen when solving this problem with $tol = 10^{-4}$ is $h = 1.8408917$. For this chosen we have

$$\delta_1(\tau) = -0.0003209724071\tau^6 - 0.002695685668\tau^5 + 0.00932492818\tau^4 - 0.009600799981\tau^3 + 0.003550042023\tau^2 - 0.000257512166\tau.$$

We plot the exact defect $\delta_1(\tau)$ in Figure 2.6. The shape of $\delta_1(\tau)$ is similar to $q_1(\tau)$. The roots of $\delta_1(\tau)$ are $0, 0.09456010641, 0.9999997033, 0.8686865028 + 0.2947534967e - 1 * I, -11.23042941, 0.8686865028 - 0.2947534967e - 1 * I$. Obviously, the roots of $\delta_1(\tau)$ for $h = 1.8408917$ and the roots of $q_1(\tau)$ are close. Therefore, we can conclude that the shape of $q_1(\tau)$ dominates the shape of defect curve $\delta_1(\tau)$ for a reasonable range of stepsize.

2.5 Implementation of the 4/5

The shape of defect curves should be consistent with the $q_1(\tau)$ as $h \to 0$. The local maximum of the defect should occur close to the point where the maximum of $q_1(\tau)$ occurs. For our new CRK45 we know the maximum of $q_1(\tau)$ occurs at $\tau* = 0.3891$, where $\tau \in [0, 1]$, from the previous section. Therefore, we evaluate the defect δ at $\tau* = 0.3891$ as the estimated maximum defect. If the evaluated maximum defect is within the required tolerance *tol*, we accept the step; otherwise, we reduce the step size h. The pseudo-code for CRK45 is shown below:

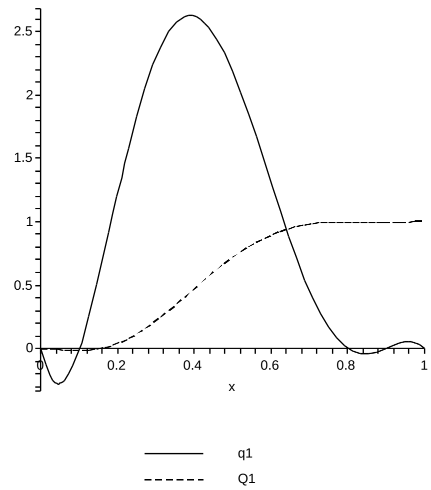

Figure 2.4: The plot of q_1 and Q_1. The q_1 is presented by the solid line, and the Q_1 is presented by the dash line.

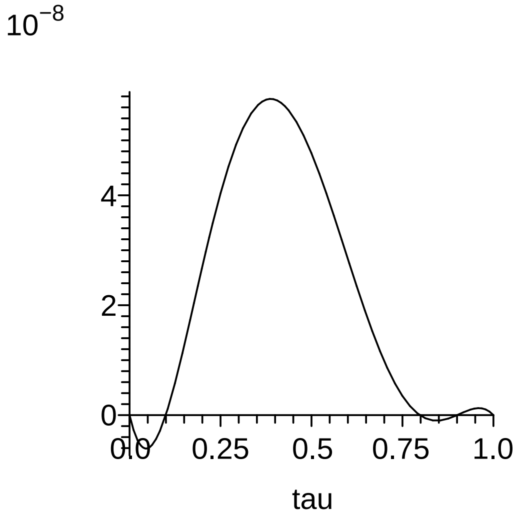

Figure 2.5: Plot of $\delta_1(\tau)$ for $h = 0.15773933612005$ ($tol = 10^{-6}$).

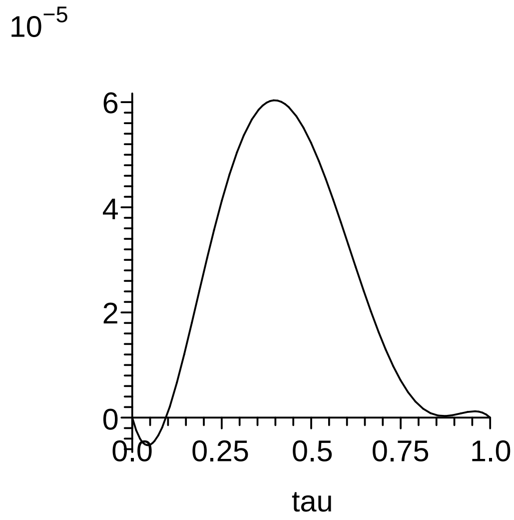

Figure 2.6: Plot of $\delta_1(\tau)$ for $h = 1.8408917$ ($tol = 10^{-4}$).

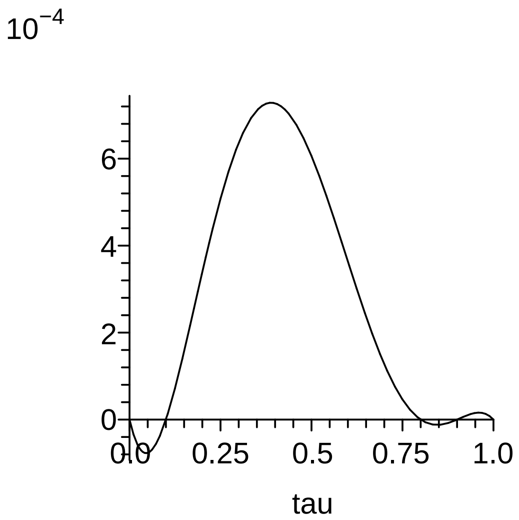

Figure 2.7: The plot of $q_1(\tau)$

CRK45()

initialize $h, t_0, y_0, t_{end}, tol$
$i \leftarrow 1$
$t_i \leftarrow t_0$
while $t_i < t_{end}$
  do APFORM()
    $\delta_i(\tau*) \leftarrow$ DEFECT()
    if $\|\delta_i(\tau*)\| < tol$
      then $i \leftarrow i + 1$
        $t_i \leftarrow t_{i-1} + h$
      else reduce h and try again

Ideally, for most of the steps, the true maximum defect will occur close to $\tau*$. But from our experimental results, we notice that for a few steps of some problems, the observed true maximum defect occurs at points that are not close to $\tau*$. In particular, this can happen when h is not small. We can detect and handle this with four extra sample points to monitor the defect. First, two points are chosen, for our formula, $\tau_1 = 0.2069$ and $\tau_2 = 0.5997$, at which the value of $q_1(\tau_1)$ and $q_1(\tau_2)$ are half of $q_1(\tau*)$. If the ratios of $\frac{\delta(\tau_1)}{\delta(\tau*)}$ and $\frac{\delta(\tau_2)}{\delta(\tau*)}$ are both close to $\frac{1}{2}$, we accept the defect $\delta(\tau*)$ as an estimated maximum defect; otherwise, we assume we do not have a small enough step size for the asymptotic analysis to be relevant and we perform another two additional defect evaluations. These two additional points are at $\tau_3 = 0.2632$ and $\tau_4 = 0.5274$. Finally, choose the maximum value among $\delta(\tau*)$, $\delta(\tau_1)$, $\delta(\tau_2)$, $\delta(\tau_3)$, and $\delta(\tau_4)$ as an estimated maximum defect value. The pseudo-code with this validity check algorithm is shown below:

CRK45()

initialize $h, t_0, y_0, t_{end}, tol$
$i \leftarrow 1$
$t_i \leftarrow t_0$
while $t_i < t_{end}$
    do APFORM()
        $\delta_i(\tau*) \leftarrow$ DEFECT()
        ▷ Validity check
        $\delta_i(\tau_1) \leftarrow$ DEFECT()
        $\delta_i(\tau_2) \leftarrow$ DEFECT()
        $R_1 \leftarrow \frac{\delta_i(\tau_1)}{\delta_i(\tau*)}$
        $R_2 \leftarrow \frac{\delta_i(\tau_2)}{\delta_i(\tau*)}$
        if $|R_1 - 0.5| < 0.2$ and $|R_2 - 0.5| < 0.2$
            then $\delta_i(\tau_3) \leftarrow$ DEFECT()
                $\delta_i(\tau_4) \leftarrow$ DEFECT()
                $\delta_i(\tau*) \leftarrow \textsc{max}(\delta_i(\tau*), \delta_i(\tau_1), \delta_i(\tau_2), \delta_i(\tau_3), \delta_i(\tau_4))$
        ▷ End of validity check
        if $\delta_i(\tau*) < tol$
            then $i \leftarrow i + 1$
                $t_i \leftarrow t_{i-1} + h$
            else reduce h and try again

The Fortran source code for our experimental implementation of this CRK45 is included in Appendix B.2, and it also available as `crk45exp.f` which can be downloaded from our website.

2.6 Shape of Defect Curves

In this section, we plot the shape of defect curves and compare with the previous work. After investigating most of the problems in the non-stiff package [7], we pick a typical problem A3

$$y' = y\cos t, \quad y(0) = 1, \quad t \in [0, 20].$$

Figure 2.8 shows the solution of A3. Figure 2.9 shows the normalized defect across each step for all 68 steps for the degree 5 improved 4/5 pair suggested in Enright and Hayes in [5]. These are steps used in solving the problem with $tol = 10^{-6}$. Figure 2.10 and Figure 2.11 plot the corresponding normalized defect curves of A3 for all 70 steps for the interpolant $v_i(t)$ without and with the validity check discussed earlier. The shape of defect curves is more consistent in Figure 2.10 compared with the ones in Figure 2.9. The true maximum defects in Figure 2.10 and Figure 2.11 occur within the range of [0.2, 0.6].

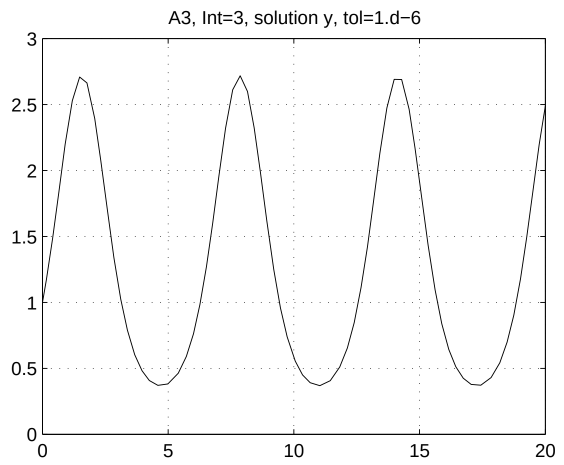

Figure 2.8: The plot of the solution of the problem A3.

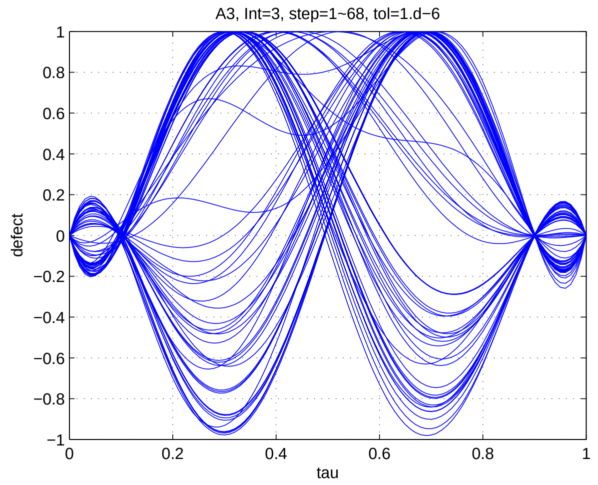

Figure 2.9: The plot of the normalized defect across a step for all steps of the problem A3 for the interpolant for the 4/5 pair in [5].

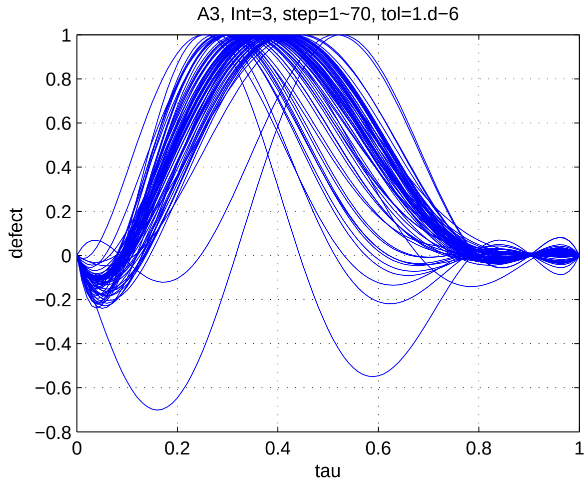

Figure 2.10: Same as Figure 2.9, but for the new interpolant $v_i(t)$ presented in this thesis.

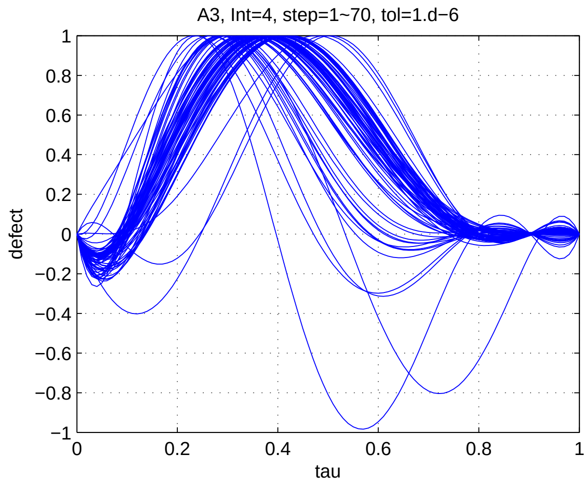

Figure 2.11: Same as Figure 2.10, but for the new interpolant $v_i(t)$ with validity check presented in this thesis.

2.7 Numerical Results

In this section we present a summary of a detailed performance analysis of the CRK45 methods investigated in this chapter. We use a modified version of the DETEST Test package [7] to assess performance on 25 non-stiff problems over a range of tolerance from 10^{-1} to 10^{-9}. This package will monitor the solution of all problems at all accuracy requirements and report measures of how well the size of the defect was kept less than the tolerance and at what cost. In addition, the package is used to determine how well the defect estimate is able to reflect the actual maximum size of the defect on each step.

The implementations investigated are five methods: Relaxed Defect Control (RDC) refers to the use of the interpolant $u_i(t)$; Strict Defect Control (SDC) and Strict Defect Control with Validity Check (SDCV) in Table 2.2 refer to the use of the interpolant $v_i(t)$ presented by Enright and Hayes in 2007 [5]; SDC and SDCV in Table 2.3 refer to use of the interpolant $v_i(t)$ defined in this thesis.

The statistics reported for each problem are Number of Steps (NSTP) and Number of Function Evaluation (NFCN) that indicate how efficient the method is. DMAX is the maximum of the ratio of the maximum observed defect to tol, and the value of DMAX

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
10^{-2}	RDC	609	7153	2.373	0.199	18.852	0.182
	SDC	613	8913	11.475	0.018	56.634	0.378
	SDCV	618	11100	1.023	0.002	1.826	0.510
10^{-4}	RDC	1070	12130	5.886	0.179	126.817	0.135
	SDC	1044	14633	5.313	0.079	60.024	0.310
	SDCV	1076	19018	1.039	0.008	1.088	0.466
10^{-6}	RDC	2176	23146	5.444	0.233	55.436	0.091
	SDC	2039	26854	33.432	0.222	91.214	0.319
	SDCV	2204	36905	1.044	0.008	1.085	0.417
10^{-8}	RDC	4929	46051	21.277	0.354	207.400	0.073
	SDC	4421	52022	24.893	0.324	723.148	0.355
	SDCV	5051	74342	1.043	0.004	1.084	0.400

Table 2.2: Results on the 25 DETEST Test Problems for the 4/5 (SDC and SDCV refer to the CRK45 method presented in [5]) with abscissa = [0.1, 0.9].

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
	RDC	609	7153	2.373	0.199	18.852	0.182
10^{-2}	SDC	623	9853	1.018	0.003	8.123	0.631
	SDCV	625	11709	0.971	0.000	1.053	0.675
	RDC	1070	12130	5.886	0.179	126.817	0.135
10^{-4}	SDC	1065	16081	1.604	0.005	7.115	0.733
	SDCV	1065	19033	1.010	0.001	1.118	0.776
	RDC	2176	23146	5.444	0.233	55.436	0.091
10^{-6}	SDC	2095	30037	1.436	0.007	11.487	0.828
	SDCV	2099	35703	1.012	0.002	1.083	0.856
	RDC	4929	46051	21.277	0.354	207.400	0.073
10^{-8}	SDC	4562	56953	1.241	0.003	32.804	0.937
	SDCV	4566	66937	1.008	0.001	1.065	0.946

Table 2.3: Results on the 25 DETEST Test Problems for the 4/5 (SDC and SDCV refer to the CRK45 method presented in this thesis, the improved interpolant $v_i(t)$), and abscissa = [0.1, 0.8, 0.9].

should be close to 1; Frac-D is the fraction of steps where DMAX is greater than 1, and Frac-D should be close to 0. Both DMAX and Frac-D reflect how reliably a method controls the maximum magnitude of the defect. The value of R-Max and Frac-G reflect how well the estimate of the max magnitude of the defect is able to reflect its true value. The Detest package determines the "true" maximum defect by sampling at 100 equally spaced points per step. The value of R-Max is the ratio of the true maximum defect (over an entire step) to the estimated maximum defect; Frac-G is defined to be the fraction of steps where $RMAX < 1.01$ and is therefore equal to the fraction of steps where the defect estimate is within 1% of the true maximum defect.

From the results in Table 2.2 and Table 2.3, we conclude that the maximum defect across entire step can be reliably controlled using our new improved CRK45, the new interpolant $v_i(t)$. It must be acknowledged that one more extra function evaluation is required for each step compared with the interpolant $v_i(t)$ of [5]. The detailed numerical results of SDC for nine different tolerances from 10^{-1} to 10^{-9} have been included in Appendix C.1 and Appendix C.2.

Chapter 3

The 5/6 Pair

3.1 Standard 6^{th}-order Interpolating Polynomial

The standard Continuous Runge-Kutta 5/6 (CRK56) method is implemented based on [6] and [4], which is the 6^{th}-order one-step interpolant associated with Verner's class of 8-stage, 6^{th}-order formula. The extra evaluation of the solution at the midpoint of each step and its derivative are used to determine the interpolant formula. Since the defect approach only uses the 6^{th}-order formula for the standard CRK56, and the 6^{th}-order formula does not use the 6^{th} stage, we develop the improved CRK56 without using the 6^{th} stage. Table 3.1 specifies this 5/6 discrete formula and one-step interpolant scheme. Since we ignore the 6^{th} stage, the exact number of stages is 8.

The intermediate approximated solution $y_{i-1+.5}$ is

$$y_{i-1+.5} = \tilde{y}_{i-1} + h \sum_{j=1}^{8} a_{\bar{s}+1,j} k_j. \tag{3.1}$$

We have

$$k_9 = y'_{i-1+.5} = f(t_{i-1} + .5h, y_{i-1+.5}), \tag{3.2}$$

0	0								
$\frac{1}{6}$	$\frac{1}{6}$	0							
$\frac{4}{15}$	$\frac{4}{75}$	$\frac{16}{75}$	0						
$\frac{2}{3}$	$\frac{5}{6}$	$-\frac{8}{3}$	$\frac{5}{2}$	0					
$\frac{5}{6}$	$-\frac{165}{64}$	$\frac{55}{6}$	$-\frac{415}{64}$	$\frac{85}{96}$	0				
$\frac{1}{15}$	$-\frac{8263}{15000}$	$\frac{124}{75}$	$-\frac{643}{680}$	$-\frac{81}{250}$	$\frac{2484}{10625}$	0			
1	$\frac{3501}{1720}$	$-\frac{300}{43}$	$\frac{297275}{52632}$	$-\frac{319}{2322}$	$\frac{24068}{84065}$	$\frac{3850}{26703}$	0		
1	$\frac{3}{40}$	0	$\frac{875}{2244}$	$\frac{23}{72}$	$\frac{264}{1955}$	$\frac{125}{11592}$	$\frac{43}{616}$	0	
$\frac{1}{2}$	$\frac{49}{640}$	0	$\frac{4375}{11968}$	$\frac{23}{384}$	$-\frac{33}{1955}$	$\frac{125}{8832}$	$-\frac{43}{1408}$	$\frac{1}{32}$	0
	$\frac{3}{40}$	0	$\frac{875}{2244}$	$\frac{23}{72}$	$\frac{264}{1955}$	$\frac{125}{11592}$	$\frac{43}{616}$	0	0

Table 3.1: The tableau of the 5/6.

Using the value $\{y_{i-1}, y'_{i-1}, y_i, y'_i, y_{i-1+.5}, y'_{i-1+.5}\}$, we determine the two stages:

$$k_{10} = f(t_{i-1} + .90h, z_i(t_{i-1} + .90h)), \tag{3.3}$$

$$k_{11} = f(t_{i-1} + .95h, z_i(t_{i-1} + .95h)), \tag{3.4}$$

where the interpolant is

$$\begin{aligned}
z_i(t_{i-1} + .90h) &= \frac{128}{3125} y_{i-1} + \frac{2592}{3125} y_i + \frac{81}{625} y_{i-1+.5} \\
&\quad + h(\frac{18}{3125} y'_{i-1} - \frac{162}{3125} y'_i + \frac{162}{3125} y'_{i-1+.5}), \\
z_i(t_{i-1} + .95h) &= \frac{5427}{400000} y_{i-1} + \frac{380133}{400000} y_i + \frac{361}{10000} y_{i-1+.5} \\
&\quad + h(\frac{1539}{800000} y'_{i-1} - \frac{29241}{800000} y'_i + \frac{3249}{200000} y'_{i-1+.5}).
\end{aligned}$$

Then the 6^{th}-order interpolant $u_i(t)$ is defined with 11 stages as

$$u_i(t_{i-1} + \tau h) = \tilde{y}_{i-1} + h \sum_{j=1}^{11} \bar{b}_j(\tau) k_j, \tag{3.5}$$

where

$$\begin{bmatrix} \bar{b}_1(\tau) \\ \bar{b}_2(\tau) \\ \bar{b}_3(\tau) \\ \bar{b}_4(\tau) \\ \bar{b}_5(\tau) \\ \bar{b}_6(\tau) \\ \bar{b}_7(\tau) \\ \bar{b}_8(\tau) \\ \bar{b}_9(\tau) \\ \bar{b}_{10}(\tau) \\ \bar{b}_{11}(\tau) \end{bmatrix} = \begin{bmatrix} 1 & -\frac{553471}{116280} & \frac{1869403}{174420} & -\frac{4243}{340} & \frac{106027}{14535} & -\frac{14810}{8721} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{149625}{12716} & -\frac{772625}{19074} & \frac{361375}{6358} & -\frac{117250}{3179} & \frac{87500}{9537} \\ 0 & \frac{1311}{136} & -\frac{20309}{612} & \frac{9499}{204} & -\frac{1541}{51} & \frac{1150}{153} \\ 0 & \frac{135432}{33235} & -\frac{466224}{33235} & \frac{654192}{33235} & -\frac{424512}{33235} & \frac{21120}{6647} \\ 0 & \frac{7125}{21896} & -\frac{110375}{98532} & \frac{7375}{4692} & -\frac{8375}{8211} & \frac{6250}{24633} \\ 0 & \frac{22059}{10472} & -\frac{37969}{5236} & \frac{7611}{748} & -\frac{8643}{1309} & \frac{2150}{1309} \\ 0 & \frac{171}{17} & -\frac{4673}{51} & \frac{3852}{17} & -\frac{3732}{17} & \frac{3800}{51} \\ 0 & -\frac{551}{34} & \frac{31421}{459} & -\frac{33151}{306} & \frac{11648}{153} & -\frac{9200}{459} \\ 0 & \frac{2375}{306} & -\frac{57125}{459} & \frac{11375}{34} & -\frac{50000}{153} & \frac{50000}{459} \\ 0 & -\frac{8000}{323} & \frac{2032000}{8721} & -\frac{88000}{153} & \frac{1600000}{2907} & -\frac{1600000}{8721} \end{bmatrix} \begin{bmatrix} \tau \\ \tau^2 \\ \tau^3 \\ \tau^4 \\ \tau^5 \\ \tau^6 \end{bmatrix}.$$

3.2 Derivative of $q_1(t)$ for the 5/6

In [5], the abscissa vector of the 5^{th}-degree of Lagrange polynomial form is [0.1, 0.8, 0.9]. For our new improved CRK56 method, we choose a 6^{th}-degree Lagrange polynomial to eliminate the interpolation error. We introduce the abscissa vector, $[c_{12}, c_{13}, c_{14}, c_{15}]$. In

Maple, we define

$Q_1(x) = b_{10} + b_{11}x + b_{12}x^2 + b_{13}x^3 + b_{14}x^4 + b_{15}x^5 + b_{16}x^6 + b_{17}x^7;$

$q_1(x) = b_{11} + 2b_{12}x + 3b_{13}x^2 + 4b_{14}x^3 + 5b_{15}x^4 + 6b_{16}x^5 + 7b_{17}x^6;$

solve($\{Q_1(0) = 0, Q_1(1) = 1, q_1(0) = 0, q_1(1) = 0, q_1(c_{12}) = 0, q_1(c_{13}) = 0, q_1(c_{14}) = 0,$

$q_1(c_{15}) = 0\}, \{b_{10}, b_{11}, b_{12}, b_{13}, b_{14}, b_{15}, b_{16}, b_{17}\}$);

...

$Q_7(x) = b_{70} + b_{71}x + b_{72}x^2 + b_{73}x^3 + b_{74}x^4 + b_{75}x^5 + b_{76}x^6 + b_{77}x^7;$

$q_7(x) = b_{71} + 2b_{72}x + 3b_{73}x^2 + 4b_{74}x^3 + 5b_{75}x^4 + 6b_{76}x^5 + 7b_{77}x^6;$

solve($\{Q_7(0) = 0, Q_7(1) = 0, q_7(0) = 0, q_7(1) = 0, q_7(c_{12}) = 0, q_7(c_{13}) = 0, q_7(c_{14}) = 0,$

$q_7(c_{15}) = 1\}, \{b_{70}, b_{71}, b_{72}, b_{73}, b_{74}, b_{75}, b_{76}, b_{77}\}$);

The criteria to choose c_{12}, c_{13}, c_{14}, and c_{15} are the same as we described in 2.2. Based on several experimental results, we choose [0.07, 0.14, 0.86, 0.93] as the abscissa vector. The plot of q_1 and Q_1 is shown in Figure 3.1. All plots of q_j are shown in Figure 3.2. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.67. The maximum of q_1 occurs at $\tau^* = 0.5$.

3.3 Formula of the New 5/6

To eliminate the interpolation error, for the new improved CRK56, we have introduced one more point to the abscissa vector. We compute k_{12}, k_{13}, k_{14}, and k_{15} corresponding to the abscissa vector:

$$k_{12} = f(t_{i-1} + 0.07h, u_i(t_{i-1} + 0.07h)), \tag{3.6}$$

$$k_{13} = f(t_{i-1} + 0.14h, u_i(t_{i-1} + 0.14h)), \tag{3.7}$$

$$k_{14} = f(t_{i-1} + 0.86h, u_i(t_{i-1} + 0.86h)), \tag{3.8}$$

$$k_{15} = f(t_{i-1} + 0.93h, u_i(t_{i-1} + 0.93h)). \tag{3.9}$$

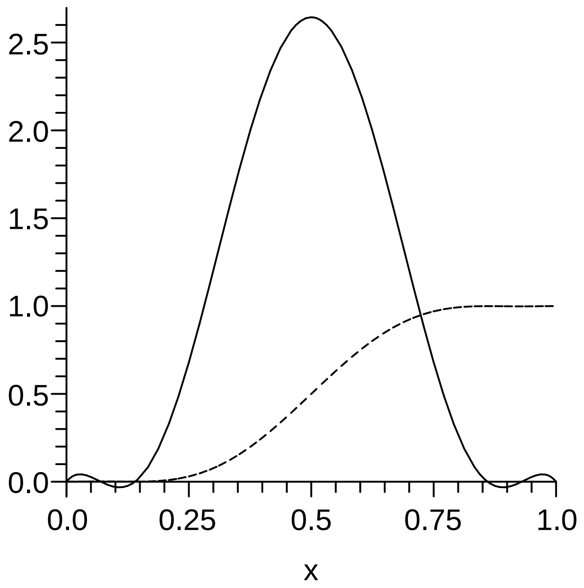

Figure 3.1: The plot of q_1 and Q_1. The dash line represents q_1, and solid line represents the Q_1. The maximum of q_1 occurs at $\tau * = 0.5, \tau \in [0, 1]$

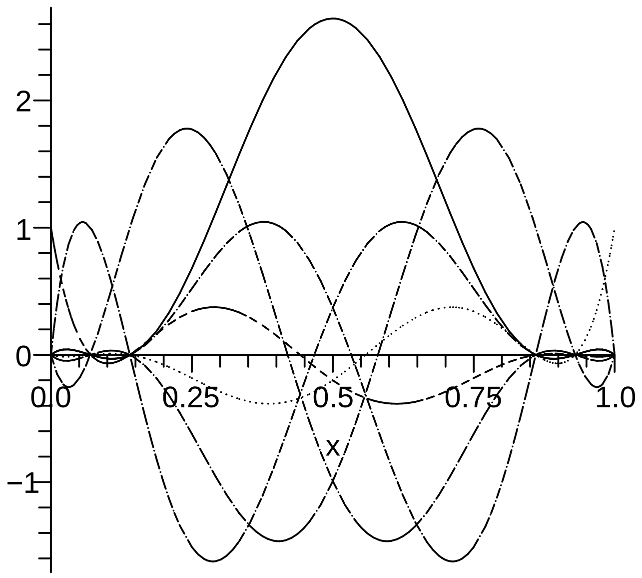

Figure 3.2: The plot of $q_1 \sim q_7$. The q_1 is presented by the solid line and has the highest magnitude among all q_j. The abscissa vector is [0.07, 0.14, 0.86, 0.93]. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.67.

The 7^{th}-order new interpolating polynomial can be written as

$$v_i(t_{i-1}+\tau h) = Q_0(\tau)y_{i-1}+Q_1(\tau)y_i+Q_2(\tau)hy'_{i-1}+Q_3(\tau)hy'_i+h\sum_{j=1}^{4} Q_{3+j}(\tau)k_{11+j}. \quad (3.10)$$

Since we know

$$Q_0(\tau) = 1 - Q_1(\tau), \quad (3.11)$$

$$y_i = y_{i-1} + h\sum_{j=1}^{8} w_j k_j, \quad (3.12)$$

$$y'_{i-1} = k_1, \quad (3.13)$$

$$y'_i = k_8, \quad (3.14)$$

substitute (3.11), (3.12), (3.13), and (3.14) into (3.10), and we have

$$\begin{aligned} v_i(t_{i-1}+\tau h) &= (1-Q_1(\tau))y_{i-1} + Q_1(\tau)(y_{i-1} + h\sum_{j=1}^{7} w_j k_j) + Q_2(\tau)hk_1 + Q_3(\tau)hk_8 \\ &\quad + h\sum_{j=1}^{4} Q_{3+j}(\tau)k_{11+j} \\ &= y_{i-1} + h\sum_{j=1}^{7} w_j Q_1(\tau)k_j + hQ_2(\tau)k_1 + hQ_3(\tau)k_8 + h\sum_{j=1}^{4} Q_{3+j}(\tau)k_{11+j} \\ &= y_{i-1} + h(w_1Q_1(\tau)+Q_2(\tau))k_1 + h\sum_{j=2}^{7} w_j Q_1(\tau)k_j + hQ_3(\tau)k_8 \\ &\quad + h\sum_{j=1}^{4} Q_{3+j}(\tau)k_{11+j}, \end{aligned}$$

which can be re-written as

$$v_i(t_{i-1}+\tau h) = y_{i-1} + h\sum_{j=1}^{15} \hat{b}_j(\tau)k_j, \quad (3.15)$$

where

$$\begin{cases} \hat{b}_1 = w_1 Q_1(\tau) + Q_2(\tau) \\ \hat{b}_2 = w_2 Q_1(\tau) \\ \hat{b}_3 = w_3 Q_1(\tau) \\ \hat{b}_4 = w_4 Q_1(\tau) \\ \hat{b}_5 = w_5 Q_1(\tau) \\ \hat{b}_6 = w_6 Q_1(\tau) \\ \hat{b}_7 = w_7 Q_1(\tau) \\ \hat{b}_8 = Q_3(\tau) \\ \hat{b}_9 = 0 \\ \hat{b}_{10} = 0 \\ \hat{b}_{11} = 0 \\ \hat{b}_{12} = Q_4(\tau) \\ \hat{b}_{13} = Q_5(\tau) \\ \hat{b}_{14} = Q_6(\tau) \\ \hat{b}_{15} = Q_7(\tau) \end{cases}$$

The $Q_1 \sim Q_7$ can be easily solved in Maple. The corresponding coefficient $\hat{b}_j$ is defined by

$$\hat{b}_j = \sum_{k=1}^{7} \hat{\beta}_{jk} \tau^k, \tag{3.16}$$

where $\hat{\beta}_{jk}$ is given in Appendix A.1.

3.4 Shape of Defect Curves

As in Chapter 2.6, we choose the problem A3:

$$y' = y \cos t, \quad y(0) = 1, \quad t \in [0, 20].$$

Figure 3.3 shows the normalized defect across a step for all steps for the improved 5/6 pair in [5]. Figure 3.4 and Figure 3.5 plot normalized defect curves of A3 for all steps for the interpolant $v_i(t)$ without and with validity check presented in this thesis. Obviously, the shape of defect curves is significantly more consistent in Figure 3.4 and Figure 3.5 compared with the one in Figure 3.3. The true maximum defects in Figure 3.4 and Figure 3.5 occur within the range of [0.45, 0.55].

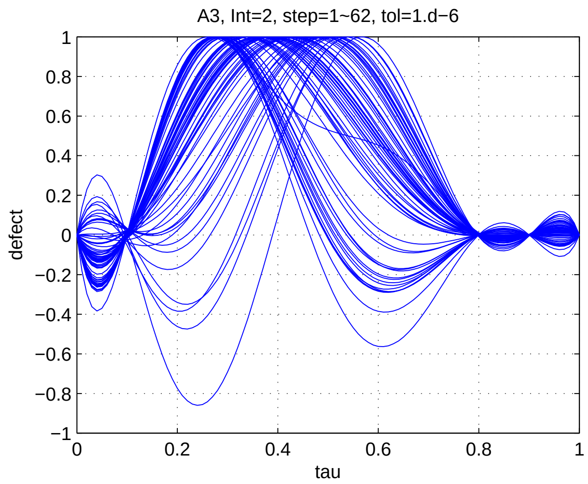

Figure 3.3: The plot of the normalized defect across a step for all steps of the example problem A3 for the interpolant $v_i(t)$ for the 5/6 pair in Enright's paper [5].

3.5 Numerical Results

As in Chapter 2.7, we use the modified version of DETEST Test package [7] to assess performance on 25 non-stiff problems over a range of tolerances from 10^{-1} to 10^{-9}. Five methods are investigated: RDC refers to the use of the interpolant $u_i(t)$; SDC and SDCV

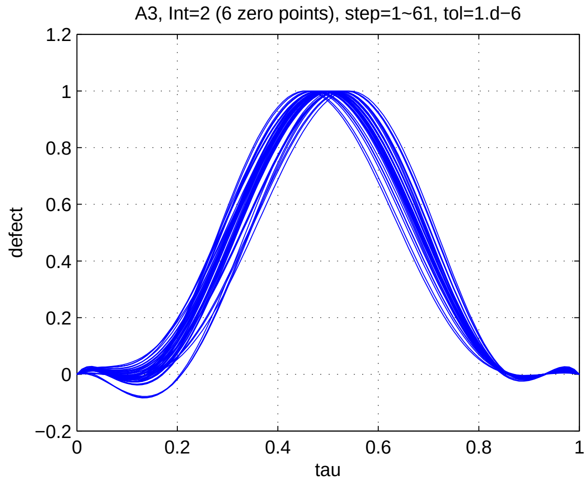

Figure 3.4: Same as Figure 3.3, but for the new interpolant $v_i(t)$ presented in this thesis.

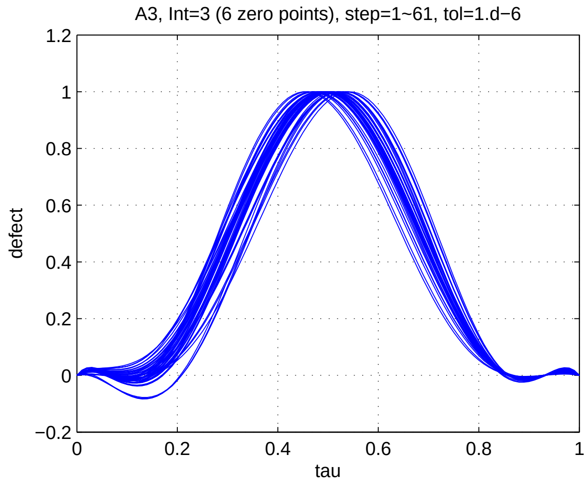

Figure 3.5: Same as Figure 3.4, but for the new interpolant $v_i(t)$ with validity check presented in this thesis.

in Table 3.2 refer to the use of the interpolant $v_i(t)$ presented by Enright and Hayes in 2007 [5]; SDC and SDCV in Table 3.3 refer to use of the interpolant $v_i(t)$ defined in this thesis.

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
10^{-2}	RDC	552	7879	5.27	0.176	23.25	0.50
	SDC	540	9713	1.72	0.006	12.80	0.55
	SDCV	539	11299	1.02	0.004	1.07	0.60
10^{-4}	RDC	955	13082	4.87	0.114	15.34	0.55
	SDC	928	16209	2.17	0.009	28.16	0.67
	SDCV	930	18885	1.04	0.002	1.10	0.70
10^{-6}	RDC	1789	23499	10.74	0.103	112.90	0.59
	SDC	1749	29019	4.38	0.009	39.25	0.67
	SDCV	1755	33885	1.02	0.003	1.11	0.71
10^{-8}	RDC	3622	43288	6.48	0.098	1286.00	0.67
	SDC	3547	53687	5.48	0.016	46.32	0.77
	SDCV	3561	62531	1.05	0.006	1.10	0.79

Table 3.2: Results on the 25 DETEST Test Problems for the 5/6 (SDC and SDCV refer to the CRK56 method presented in [5] with abscissa = [0.1, 0.8, 0.9]).

For the explanation of the notation used in Table 3.2 and Table 3.3, refer to Section 2.7. From the results in Tables 3.2 and 3.3, we conclude that the maximum defect across entire step can be reliably controlled by using our new improved CRK56, the new interpolant $v_i(t)$, although this new method requires one more extra function evaluation for each step compared with the interpolant $v_i(t)$ in [5].

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
10^{-2}	RDC	552	7879	5.272	0.176	23.253	0.498
	SDC	547	10585	0.996	0.000	1.741	0.700
	SDCV	549	12300	0.996	0.000	1.431	0.712
10^{-4}	RDC	955	13082	4.867	0.144	15.340	0.554
	SDC	929	17305	4.902	0.003	18.900	0.869
	SDCV	931	19819	1.001	0.001	1.079	0.875
10^{-6}	RDC	1789	23499	10.746	0.103	112.903	0.587
	SDC	1748	30925	1.007	0.001	1.807	0.959
	SDCV	1748	35073	1.007	0.001	1.085	0.960
10^{-8}	RDC	3622	43288	6.477	0.098	1286.697	0.672
	SDC	3547	57460	1.013	0.001	1.138	0.980
	SDCV	3547	65148	1.013	0.001	1.073	0.980

Table 3.3: Results on the 25 DETEST Test Problems for the 5/6 (SDC and SDCV refer to the CRK56 method presented in this thesis, the improved interpolant $v_i(t)$, with abscissa = [0.07, 0.14, 0.86, 0.93]).

Chapter 4

The 7/8 pair

4.1 Standard 8^{th}-order Interpolating Polynomial

The well-known tableau for the discrete 8^{th}-order, explicit, 14-stage Runge-Kutta 7/8 pair is shown in Table 4.1 and the corresponding augmented tableau is given in `crk78coef.ps` which can be downloaded separately from our website.

$$
\begin{array}{c|cccccc}
0 & 0 & & & & & \\
c_2 & a_{21} & 0 & & & & \\
c_3 & a_{31} & a_{32} & 0 & & & \\
c_4 & a_{41} & a_{42} & a_{43} & 0 & & \\
c_5 & a_{51} & a_{52} & a_{53} & a_{54} & 0 & \\
c_6 & a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & 0 \\
\dots & \dots & & & & & \\
c_{14} & a_{14,1} & a_{14,2} & \dots & & a_{14,13} & 0 \\
\hline
 & w_1 & w_2 & \dots & & & w_{14}
\end{array}
$$

Table 4.1: The notation of the tableau of the 7/8.

Since the defect approach only uses the 8^{th}-order discrete formula and the 13^{th} stage is never used in the 8^{th}-order formula, we do not compute k_{13}. The 8^{th}-order discrete solution $\tilde{y}_i$ is defined by

$$\tilde{y}_i = \tilde{y}_{i-1} + h \sum_{j=1, j\neq 13}^{14} w_j k_j, \tag{4.1}$$

where

$$k_j = f(t_{i-1} + c_j h, \tilde{Y}_j), \tag{4.2}$$

$$\tilde{Y}_j = \tilde{y}_{i-1} + h \sum_{r=1}^{j-1} a_{jr} f(t_{i-1} + c_r h, \tilde{Y}_r). \tag{4.3}$$

The standard (non-optimal) 8^{th}-order interpolating polynomial $z_i(t)$ is given by [4]

$$z_i(t) = \tilde{y}_{i-1} + h \sum_{j=1, j\neq 13}^{21} b_j(\tau) k_j, \quad \tau \in [t_{i-1}, t_{i+1}], \tag{4.4}$$

where

$$b_j = \sum_{k=1}^{8} \beta_{j,k} \tau^k,$$

and the corresponding augmented tableau of $\beta_{j,k}$ is included in `crk78coef.ps`. Since the 8^{th}-order interpolant $z_i(t)$ needs additional f-evaluations, the corresponding augmented tableau of c_j and a_{jr} is also included in `crk78coef.ps`. The k_j are computed by (4.2) and (4.3) using this extended tableau of c_j and a_{jr}.

4.2 Derivative of $q_1(t)$ for the 7/8

In [5], the abscissa vector defining the 8^{th}-degree Lagrange polynomial form is [0.07, 0.18, 0.73, 0.82, 0.93]. For our new improved Continuous Runge-Kutta 7/8 method (CRK78), we introduce a 9^{th}-degree Lagrange polynomial to eliminate the interpolation error. Assume the unique polynomial associated with the abscissa vector, $[c_{24}, c_{25}, \ldots, c_{29}]$. In

Maple, we define

$$Q_1(x) = b_{10} + b_{11}x + b_{12}x^2 + b_{13}x^3 + b_{14}x^4 + b_{15}x^5 + b_{16}x^6 + b_{17}x^7 + b_{18}x^8 + b_{19}x^9;$$

$$q_1(x) = b_{11} + 2b_{12}x + 3b_{13}x^2 + 4b_{14}x^3 + 5b_{15}x^4 + 6b_{16}x^5 + 7b_{17}x^6 + 8b_{18}x^7 + 9b_{19}x^8;$$

$$\text{solve}(\{Q_1(0) = 0, Q_1(1) = 1, q_1(0) = 0, q_1(1) = 0, q_1(c_{24}) = 0, q_1(c_{25}) = 0, q_1(c_{26}) = 0,$$

$$q_1(c_{27}) = 0, q_1(c_{28}) = 0, q_1(c_{29}) = 0\}, \{b_{10}, b_{11}, b_{12}, b_{13}, b_{14}, b_{15}, b_{16}, b_{17}, b_{18}, b_{19}\});$$

$$\dots$$

$$Q_9(x) = b_{90} + b_{91}x + b_{92}x^2 + b_{93}x^3 + b_{94}x^4 + b_{95}x^5 + b_{96}x^6 + b_{97}x^7 + b_{98}x^8 + b_{99}x^9;$$

$$q_9(x) = b_{91} + 2b_{92}x + 3b_{93}x^2 + 4b_{94}x^3 + 5b_{95}x^4 + 6b_{96}x^5 + 7b_{97}x^6 + 8b_{98}x^7 + 9b_{99}x^8;$$

$$\text{solve}(\{Q_9(0) = 0, Q_9(1) = 0, q_9(0) = 0, q_9(1) = 0, q_9(c_{24}) = 0, q_9(c_{25}) = 0, q_9(c_{26}) = 0,$$

$$q_9(c_{27}) = 0, q_9(c_{28}) = 0, q_9(c_{29}) = 1, \}, \{b_{90}, b_{91}, b_{92}, b_{93}, b_{94}, b_{95}, b_{96}, b_{97}, b_{98}, b_{99}\});$$

Solving these equations in Maple, we get all coefficients for Q_j and q_j. The criteria to choose the abscissa vector are the same as we described in Chapter 2.2. By observing several experimental results, we choose the abscissa vector to be [0.07, 0.14, 0.21, 0.79, 0.86, 0.93]. The plots of q_1 and Q_1 are shown in Figure 4.1. The plots of all q_j are shown in Figure 4.2. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.53. The maximum of q_1 occurs at $\tau^* = 0.5$, $\tau \in [0, 1]$.

4.3 Formula of the New 7/8

To eliminate interpolation error, for the new improved CRK78, we introduced one more point to the abscissa vector, which is [.07, .14, .21, .79, .86, .93] from the previous section.

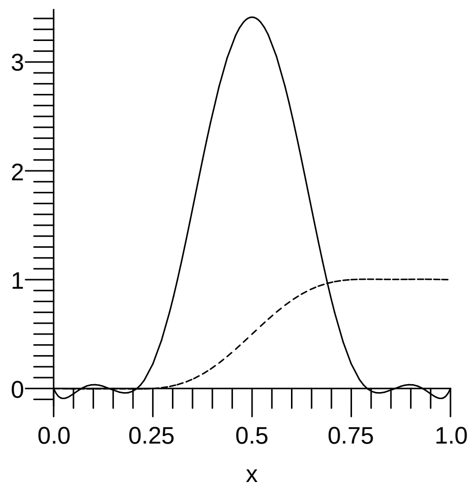

Figure 4.1: The plot of q_1 and Q_1. The dash line represents the q_1, and the solid line represents the Q_1. The maximum of q_1 occurs at $\tau* = 0.5, \tau \in [0, 1]$

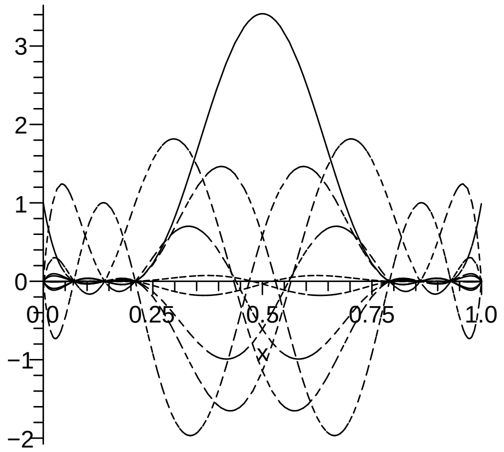

Figure 4.2: The plot of $q_1 \sim q_7$. The q_1 is presented by the solid line and has the highest magnitude among all q_j. The abscissa vector is [0.07, 0.14, 0.21, 0.79, 0.86, 0.93]. The ratio of maximum of $q_{j,j\neq 1}$ and q_1 is 0.53.

We compute k_{24}, k_{25}, k_{26}, k_{27}, k_{28}, and k_{29} corresponding to the abscissa vector

$$k_{24} = f(t_{i-1} + .07h, z_i(t_{i-1} + .07h)), \tag{4.5}$$

$$k_{25} = f(t_{i-1} + .14h, z_i(t_{i-1} + .14h)), \tag{4.6}$$

$$k_{26} = f(t_{i-1} + .21h, z_i(t_{i-1} + .21h)), \tag{4.7}$$

$$k_{27} = f(t_{i-1} + .79h, z_i(t_{i-1} + .79h)), \tag{4.8}$$

$$k_{28} = f(t_{i-1} + .86h, z_i(t_{i-1} + .86h)), \tag{4.9}$$

$$k_{29} = f(t_{i-1} + .93h, z_i(t_{i-1} + .93h)), \tag{4.10}$$

and then the 9^{th}-order new interpolant $v_i(t)$ can be written as

$$v_i(t_{i-1}+\tau h) = Q_0(\tau)y_{i-1}+Q_1(\tau)y_i+Q_2(\tau)hy'_{i-1}+Q_3(\tau)hy'_i+h\sum_{j=1}^{6} Q_{3+j}(\tau)k_{23+j}. \tag{4.11}$$

Since we know

$$Q_0(\tau) = 1 - Q_1(\tau), \tag{4.12}$$

$$y_i = y_{i-1} + h\sum_{j=1}^{12} w_j k_j, \tag{4.13}$$

$$y'_{i-1} = k_1, \tag{4.14}$$

$$y'_i = k_{14}, \tag{4.15}$$

and we do not use the 13^{th}-stage, substitute (4.12), (4.13), (4.14), and (4.15) into (4.11), and we have

$$\begin{aligned}
v_i(t_{i-1}+\tau h) &= (1-Q_1(\tau))y_{i-1} + Q_1(\tau)(y_{i-1} + h\sum_{j=1}^{12} w_j k_j) + Q_2(\tau)hk_1 + Q_3(\tau)hk_8 \\
&\quad + h\sum_{j=1}^{6} Q_{3+j}(\tau)k_{23+j} \\
&= y_{i-1} + h\sum_{j=1}^{12} w_j Q_1(\tau)k_j + hQ_2(\tau)k_1 + hQ_3(\tau)k_8 + h\sum_{j=1}^{6} Q_{3+j}(\tau)k_{23+j} \\
&= y_{i-1} + h(w_1Q_1(\tau) + Q_2(\tau))k_1 + h\sum_{j=2}^{12} w_j Q_1(\tau)k_j + hQ_3(\tau)k_8 \\
&\quad + h\sum_{j=1}^{6} Q_{3+j}(\tau)k_{23+j},
\end{aligned}$$

which can be re-written as

$$v_i(t_{i-1}+\tau h) = y_{i-1} + h\sum_{j=1}^{29} \hat{b}_j(\tau)k_j, \tag{4.16}$$

where

$$
\begin{cases}
\hat{b}_1 = w_1 Q_1(\tau) + Q_2(\tau) \\
\hat{b}_2 = w_2 Q_1(\tau) \\
\hat{b}_3 = w_3 Q_1(\tau) \\
\hat{b}_4 = w_4 Q_1(\tau) \\
\hat{b}_5 = w_5 Q_1(\tau) \\
\hat{b}_6 = w_6 Q_1(\tau) \\
\hat{b}_7 = w_7 Q_1(\tau) \\
\hat{b}_8 = w_8 Q_1(\tau) \\
\hat{b}_9 = w_9 Q_1(\tau) \\
\hat{b}_{10} = w_{10} Q_1(\tau) \\
\hat{b}_{11} = w_{11} Q_1(\tau) \\
\hat{b}_{12} = w_{12} Q_1(\tau) \\
\hat{b}_{13} = 0 \\
\hat{b}_{14} = Q_3(\tau) \\
\hat{b}_{15} = 0 \\
\quad \vdots \\
\hat{b}_{23} = 0 \\
\hat{b}_{24} = Q_4(\tau) \\
\hat{b}_{25} = Q_5(\tau) \\
\hat{b}_{26} = Q_6(\tau) \\
\hat{b}_{27} = Q_7(\tau) \\
\hat{b}_{28} = Q_8(\tau) \\
\hat{b}_{29} = Q_9(\tau).
\end{cases}
$$

The $Q_1 \sim Q_9$ can be easily solved in Maple. The corresponding coefficients $\hat{b}_j(\tau)$ is defined by

$$\hat{b}_j(\tau) = \sum_{k=1}^{9} \hat{\beta}_{jk}\tau^k, \tag{4.17}$$

where $\hat{\beta}_{jk}$ is given in `crk78coef.ps`, which can be downloaded from our website.

4.4 Shape of Defect Curves

As in Section 2.6, we choose the problem A3:

$$y' = y\cos t, \quad y(0) = 1, \quad t \in [0, 20].$$

Figure 4.3 shows the normalized defect across a step for all steps for the improved 7/8 pair in [5]. Figures 4.4 and 4.5 plot normalized defect curves of A3 for all steps for the new reliable CRK78 presented in this thesis without and with validity check respectively. Obviously, the shape of defect curves is more consistent in Figure 4.4 and Figure 4.5 compared with the one in Figure 4.3.

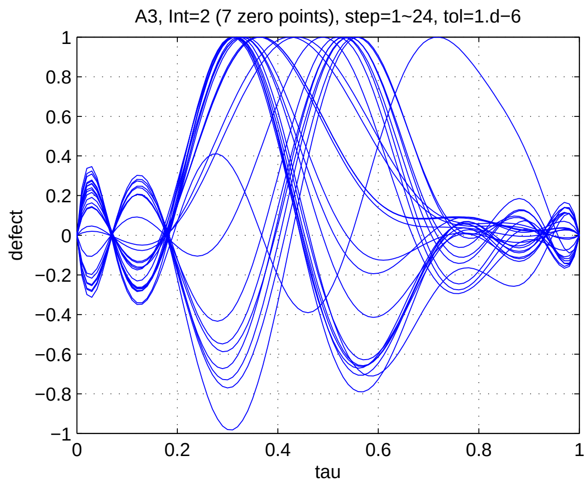

Figure 4.3: The plot of the normalized defect across a step for all steps of the problem A3 for the interpolant $v_i(t)$ for the 7/8 pair in [5].

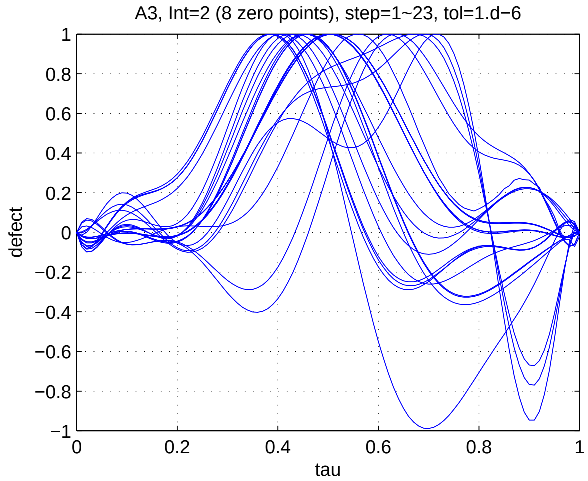

Figure 4.4: Same as Figure 4.3, but for the new interpolant $v_i(t)$ presented in this thesis.

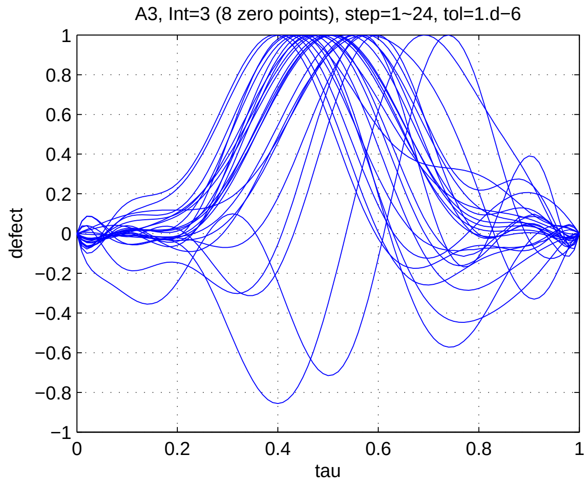

Figure 4.5: Same as Figure 4.4, but for the new interpolant $v_i(t)$ with validity check presented in this thesis.

4.5 Numerical Results

As in Chapter 2.7, we use the modified version of DETEST Test package [7] to assess performance on 25 non-stiff problems over a range of tolerance from 10^{-1} to 10^{-9}. Five methods are investigated: RDC refers to the use of the interpolant $u_i(t)$ (4.4); SDC and SDCV in Table 4.2 refer to the use of the interpolant $v_i(t)$ presented by Enright and Hayes in 2007 [5]; SDC and SDCV in Table 4.3 refer to use of the interpolant $v_i(t)$ (4.16) defined in this thesis.

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
10^{-2}	RDC	337	8745	10.684	0.213	36.707	0.299
	SDC	333	11025	1.030	0.003	44.295	0.245
	SDCV	335	12491	1.057	0.003	3.660	0.364
10^{-4}	RDC	495	13285	7.699	0.139	32.699	0.168
	SDC	489	16125	4.603	0.016	48.839	0.217
	SDCV	498	18560	0.966	0.000	3.660	0.396
10^{-6}	RDC	715	18245	6.092	0.126	134.319	0.099
	SDC	709	22775	4.859	0.051	44.433	0.240
	SDCV	738	26498	0.967	0.000	3.660	0.457
10^{-8}	RDC	1095	27065	31.124	0.179	409.091	0.077
	SDC	1082	33450	5.335	0.140	37.039	0.305
	SDCV	1490	50719	1.091	0.002	3.660	0.474

Table 4.2: Results on the 25 DETEST Test Problem for the 7/8 (SDC and SDCV refer to the CRK78 presented in [5]) with abscissa = [0.07, 0.18, 0.73, 0.82, 0.93].

For the explanation of the notation used in Table 4.2 and Table 4.3, refer to Chapter 2.7. From the results in Table 4.2 and Table 4.3, we conclude that the maximum defect across entire step can be reliably controlled by using our new improved CRK78, the new interpolant $v_i(t)$, although this new method requires one more extra function evaluation

TOL	CRK	NSTP	NFCN	DMAX	Frac-D	R-Max	Frac-G
	RDC	337	8745	10.684	0.213	36.707	0.299
10^{-2}	SDC	332	11439	7.157	0.009	30.518	0.343
	SDCV	333	12795	1.347	0.006	1.653	0.368
	RDC	495	13285	7.699	0.139	32.699	0.168
10^{-4}	SDC	466	15781	1.023	0.002	16.911	0.446
	SDCV	465	17325	1.023	0.002	2.229	0.449
	RDC	715	18245	6.092	0.126	134.319	0.099
10^{-6}	SDC	707	23425	3.014	0.008	22.719	0.576
	SDCV	712	26255	1.037	0.001	1.343	0.586
	RDC	1095	27065	31.124	0.179	409.091	0.077
10^{-8}	SDC	1081	34787	1.858	0.005	20.817	0.611
	SDCV	1081	38223	1.112	0.008	3.142	0.617

Table 4.3: Results on the 25 DETEST Test Problem for the 7/8 (SDC and SDCV refer to the CRK78 presented in this thesis, the improved interpolant $v_i(t)$), with abscissa = [0.07, 0.14, 0.21, 0.79, 0.86, 0.93].

for each step compared with the interpolant $v_i(t)$ in [5].

Chapter 5

Conclusions

5.1 Summary

In this thesis, we introduce a more robust and reliable defect control for a class of continuous Rung-Kutta methods, specially for order 5, 6, and 8. This new approach is to reduce the affects of the interpolation error by generating a higher order interpolant polynomial. Thus, for the new improved Continuous Runge-Kutta methods, the shape of defect curves across a step for all steps become more consistent, and the methods using the defect approach become more reliable and robust. Also, we demonstrate how to generate the new improved formulas of the continuous Runge-Kutta methods using defect control, and how to implement these new methods. Moreover, the plots of defect curves of an example along with the plots from the previous work are discussed. Finally, the numerical results on the 25 test problems of DETEST at a wide range of accuracy verify that our new defect control approach has significantly improved the reliability and robustness of the associated continuous Runge-Kutta method.

5.2 Future Work

This thesis investigates the development of more reliable Runge-Kutta formula for Initial Value Problem. One of the areas for future investigations is to apply the reliable formulas to the other classes of problems, such as Boundary Value Problem, Delay Differential Equations, Algebraic Differential Equations.

Other areas for future investigations could be to apply the reliable defect control using parallel processing. For example, after the discrete solution is available, one processor could complete the current step while a second processor could begin the next step (on the assumption that the current step will be accepted.)

Appendix A

The corresponding Augmented Tableau

A.1 Continuous Runge-Kutta 5/6

$\hat{\beta}_{1,1}$ = 1.0D0

$\hat{\beta}_{1,2}$ = -0.1145947870241D13 / 0.85935150840D11

$\hat{\beta}_{1,3}$ = 0.2091682624438657D16 / 0.27971891598420D14

$\hat{\beta}_{1,4}$ = -0.176420529104375D15 / 0.932396386614D12

$\hat{\beta}_{1,5}$ = 0.111605391137125D15 / 0.466198193307D12

$\hat{\beta}_{1,6}$ = -0.208482288125000D15 / 0.1398594579921D13

$\hat{\beta}_{1,7}$ = 0.17023126250000D14 / 0.466198193307D12

$\hat{\beta}_{2,1}$ = 0.0D0

$\hat{\beta}_{2,2}$ = 0.0D0

$\hat{\beta}_{2,3}$ = 0.0D0

$\hat{\beta}_{2,4}$ = 0.0D0

$\hat{\beta}_{2,5}$ = 0.0D0

$\hat{\beta}_{2,6}$ = 0.0D0

$\hat{\beta}_{2,7}$ = 0.0D0

$\hat{\beta}_{3,1}$ = 0.0D0

$\hat{\beta}_{3,2}$ = 0.38716125D8 / 0.57406756D8

$\hat{\beta}_{3,3}$ = -0.29604887375D11 / 0.2669414154D10

$\hat{\beta}_{3,4}$ = 0.26241796875D11 / 0.444902359D9

$\hat{\beta}_{3,5}$ = -0.48777968750D11 / 0.444902359D9

$\hat{\beta}_{3,6}$ = 0.38281250000D11 / 0.444902359D9

$\hat{\beta}_{3,7}$ = -0.10937500000D11 / 0.444902359D9

$\hat{\beta}_{4,1}$ = 0.0D0

$\hat{\beta}_{4,2}$ = 0.339227D6 / 0.613976D6

$\hat{\beta}_{4,3}$ = -0.778185611D9 / 0.85649652D8

$\hat{\beta}_{4,4}$ = 0.229928125D9 / 0.4758314D7

$\hat{\beta}_{4,5}$ = -0.641081875D9 / 0.7137471D7

$\hat{\beta}_{4,6}$ = 0.503125000D9 / 0.7137471D7

$\hat{\beta}_{4,7}$ = -0.143750000D9 / 0.7137471D7

$\hat{\beta}_{5,1}$ = 0.0D0

$\hat{\beta}_{5,2}$ = 0.3185784D7 / 0.13640035D8

$\hat{\beta}_{5,3}$ = -0.1624039536D10 / 0.422841085D9

$\hat{\beta}_{5,4}$ = 0.1727460000D10 / 0.84568217D8

$\hat{\beta}_{5,5}$ = -0.139608000D9 / 0.3676879D7

$\hat{\beta}_{5,6}$ = 0.2520000000D10 / 0.84568217D8

$\hat{\beta}_{5,7}$ = -0.720000000D9 / 0.84568217D8

$\hat{\beta}_{6,1}$ = 0.0D0

$\hat{\beta}_{6,2}$ = 0.263375D6 / 0.14121448D8

$\hat{\beta}_{6,3}$ = -0.604181375D9 / 0.1969941996D10

$\hat{\beta}_{6,4}$ = 0.178515625D9 / 0.109441222D9

$\hat{\beta}_{6,5}$ = - 0.21640625D8 / 0.7137471D7

$\hat{\beta}_{6,6}$ = 0.390625000D9 / 0.164161833D9

$\hat{\beta}_{6,7}$ = -0.781250000D9 / 0.1149132831D10

$\hat{\beta}_{7,1}$ = 0.0D0

$\hat{\beta}_{7,2}$ = 0.815409D6 / 0.6753736D7

$\hat{\beta}_{7,3}$ = -0.207838393D9 / 0.104682908D9

$\hat{\beta}_{7,4}$ = 0.552684375D9 / 0.52341454D8

$\hat{\beta}_{7,5}$ = -0.513661875D9 / 0.26170727D8

$\hat{\beta}_{7,6}$ = 0.403125000D9 / 0.26170727D8

$$\hat{\beta}_{7,7} = \text{-0.806250000D9 / 0.183195089D9}$$
$$\hat{\beta}_{8,1} = \text{0.0D0}$$
$$\hat{\beta}_{8,2} = \text{-0.1502282D7 / 0.2379157D7}$$
$$\hat{\beta}_{8,3} = \text{0.2141230151953D13 / 0.199799225703D12}$$
$$\hat{\beta}_{8,4} = \text{-0.28503692921875D14 / 0.466198193307D12}$$
$$\hat{\beta}_{8,5} = \text{0.20652548742500D14 / 0.155399397769D12}$$
$$\hat{\beta}_{8,6} = \text{-0.172150075000000D15 / 0.1398594579921D13}$$
$$\hat{\beta}_{8,7} = \text{0.19227575000000D14 / 0.466198193307D12}$$
$$\hat{\beta}_{9,1} = \text{0.0D0}$$
$$\hat{\beta}_{9,2} = \text{0.0D0}$$
$$\hat{\beta}_{9,3} = \text{0.0D0}$$
$$\hat{\beta}_{9,4} = \text{0.0D0}$$
$$\hat{\beta}_{9,5} = \text{0.0D0}$$
$$\hat{\beta}_{9,6} = \text{0.0D0}$$
$$\hat{\beta}_{9,7} = \text{0.0D0}$$
$$\hat{\beta}_{10,1} = \text{0.0D0}$$
$$\hat{\beta}_{10,2} = \text{0.0D0}$$
$$\hat{\beta}_{10,3} = \text{0.0D0}$$
$$\hat{\beta}_{10,4} = \text{0.0D0}$$
$$\hat{\beta}_{10,5} = \text{0.0D0}$$
$$\hat{\beta}_{10,6} = \text{0.0D0}$$
$$\hat{\beta}_{10,7} = \text{0.0D0}$$
$$\hat{\beta}_{11,1} = \text{0.0D0}$$
$$\hat{\beta}_{11,2} = \text{0.0D0}$$
$$\hat{\beta}_{11,3} = \text{0.0D0}$$
$$\hat{\beta}_{11,4} = \text{0.0D0}$$
$$\hat{\beta}_{11,5} = \text{0.0D0}$$
$$\hat{\beta}_{11,6} = \text{0.0D0}$$
$$\hat{\beta}_{11,7} = \text{0.0D0}$$
$$\hat{\beta}_{12,1} = \text{0.0D0}$$
$$\hat{\beta}_{12,2} = \text{0.27984500000D11 / 0.1315673821D10}$$
$$\hat{\beta}_{12,3} = \text{-0.19617705031000000D17 / 0.110488971813759D15}$$

$$\hat{\beta}_{12,4} = 0.19128740528500000D17 / 0.36829657271253D14$$

$$\hat{\beta}_{12,5} = -0.8683918820000000D16 / 0.12276552423751D14$$

$$\hat{\beta}_{12,6} = 0.50872142500000000D17 / 0.110488971813759D15$$

$$\hat{\beta}_{12,7} = -0.99500000000000D14 / 0.856503657471D12$$

$$\hat{\beta}_{13,1} = 0.0D0$$

$$\hat{\beta}_{13,2} = -0.2230609375D10 / 0.254646546D9$$

$$\hat{\beta}_{13,3} = 0.1242899882828125D16 / 0.10692481143267D14$$

$$\hat{\beta}_{13,4} = -0.2855923103234375D16 / 0.7128320762178D13$$

$$\hat{\beta}_{13,5} = 0.2117312366875000D16 / 0.3564160381089D13$$

$$\hat{\beta}_{13,6} = -0.4355508906250000D16 / 0.10692481143267D14$$

$$\hat{\beta}_{13,7} = 0.42156250000000D14 / 0.396017820121D12$$

$$\hat{\beta}_{14,1} = 0.0D0$$

$$\hat{\beta}_{14,2} = -0.3081078125D10 / 0.1564257354D10$$

$$\hat{\beta}_{14,3} = 0.50601484953125D14 / 0.1527497306181D13$$

$$\hat{\beta}_{14,4} = -0.1320549003015625D16 / 0.7128320762178D13$$

$$\hat{\beta}_{14,5} = 0.1373825804375000D16 / 0.3564160381089D13$$

$$\hat{\beta}_{14,6} = -0.3612022343750000D16 / 0.10692481143267D14$$

$$\hat{\beta}_{14,7} = 0.42156250000000D14 / 0.396017820121D12$$

$$\hat{\beta}_{15,1} = 0.0D0$$

$$\hat{\beta}_{15,2} = 0.1029500000D10 / 0.563860209D9$$

$$\hat{\beta}_{15,3} = -0.489308927000000D15 / 0.15784138830537D14$$

$$\hat{\beta}_{15,4} = 0.6516829271500000D16 / 0.36829657271253D14$$

$$\hat{\beta}_{15,5} = -0.14155971460000000D17 / 0.36829657271253D14$$

$$\hat{\beta}_{15,6} = 0.38976357500000000D17 / 0.110488971813759D15$$

$$\hat{\beta}_{15,7} = -0.99500000000000D14 / 0.856503657471D12$$

Appendix B

Source Code

B.1 Sample Maple Code to Solve Q_j, q_j for the 4/5

```
#################################################################################
# This is an experimental implementation to solve Lagrange polynomials Q_j and #
# q_j in Maple.                                                                 #
#################################################################################
#
# the tableau of the discrete 4/5 pair
w := [35/384,0,500/1113,125/192,-2187/6784,11/84,0];
# the abscissa vector
c1 := 10/100;
c2 := 80/100;
c3 := 90/100;
#
Q1 := x -> b10+b11*x+b12*x^2+b13*x^3+b14*x^4+b15*x^5+b16*x^6;
q1 := x -> b11+2*b12*x+3*b13*x^2+4*b14*x^3+5*b15*x^4+6*b16*x^5;
tt := solve({Q1(0)=0, Q1(1)=1, q1(0)=0, q1(1)=0, q1(c1)=0, q1(c2)=0, q1(c3)=0},
      {b10,b11,b12,b13,b14,b15,b16});
assign(tt);
Digits := 20:
qp1 := fsolve(2*b12+6*b13*x+12*b14*x^2+20*b15*x^3+30*b16*x^4 = 0,x);
```

```
#
Q2 := x -> b20+b21*x+b22*x^2+b23*x^3+b24*x^4+b25*x^5+b26*x^6;
q2 := x -> b21+2*b22*x+3*b23*x^2+4*b24*x^3+5*b25*x^4+6*b26*x^5;
tt := solve({Q2(0)=0, Q2(1)=0, q2(0)=1, q2(1)=0, q2(c1)=0, q2(c2)=0, q2(c3)=0},
          {b20,b21,b22,b23,b24,b25,b26});
assign(tt);
qp2 := fsolve(2*b22+6*b23*x+12*b24*x^2+20*b25*x^3+30*b26*x^4 = 0,x);
#
Q3 := x -> b30+b31*x+b32*x^2+b33*x^3+b34*x^4+b35*x^5+b36*x^6;
q3 := x -> b31+2*b32*x+3*b33*x^2+4*b34*x^3+5*b35*x^4+6*b36*x^5;
tt := solve({Q3(0)=0, Q3(1)=0, q3(0)=0, q3(1)=1, q3(c1)=0, q3(c2)=0, q3(c3)=0},
      {b30,b31,b32,b33,b34,b35,b36});
assign(tt);
qp3 := fsolve(2*b32+6*b33*x+12*b34*x^2+20*b35*x^3+30*b36*x^4=0, x);
#
Q4 := x -> b40+b41*x+b42*x^2+b43*x^3+b44*x^4+b45*x^5+b46*x^6;
q4 := x -> b41+2*b42*x+3*b43*x^2+4*b44*x^3+5*b45*x^4+6*b46*x^5;
tt := solve({Q4(0)=0, Q4(1)=0, q4(0)=0, q4(1)=0, q4(c1)=1, q4(c2)=0, q4(c3)=0},
       {b40,b41,b42,b43,b44,b45,b46});
assign(tt);
qp4 := fsolve(2*b42+6*b43*x+12*b44*x^2+20*b45*x^3+30*b46*x^4=0, x);
#
Q5 := x -> b50+b51*x+b52*x^2+b53*x^3+b54*x^4+b55*x^5+b56*x^6;
q5 := x -> b51+2*b52*x+3*b53*x^2+4*b54*x^3+5*b55*x^4+6*b56*x^5;
tt := solve({Q5(0)=0, Q5(1)=0, q5(0)=0, q5(1)=0, q5(c1)=0, q5(c2)=1, q5(c3)=0},
       {b50,b51,b52,b53,b54,b55,b56});
assign(tt);
qp5 := fsolve(2*b52+6*b53*x+12*b54*x^2+20*b55*x^3+30*b56*x^4=0, x);
#
Q6 := x -> b60+b61*x+b62*x^2+b63*x^3+b64*x^4+b65*x^5+b66*x^6;
q6 := x -> b61+2*b62*x+3*b63*x^2+4*b64*x^3+5*b65*x^4+6*b66*x^5;
tt := solve({Q6(0)=0, Q6(1)=0, q6(0)=0, q6(1)=0, q6(c1)=0, q6(c2)=0, q6(c3)=1},
       {b60,b61,b62,b63,b64,b65,b66});
```

```
assign(tt);
qp6 := fsolve(2*b62+6*b63*x+12*b64*x^2+20*b65*x^3+30*b66*x^4=0, x);
#
maxq1 := Optimization[Maximize](q1(x),x=0..1);
maxQ1 := Optimization[Maximize](Q1(x),x=0..1);
maxqj := max(Optimization[Maximize](q2(x),x=0..1)[1],
             Optimization[Maximize](q3(x),x=0..1)[1],
             Optimization[Maximize](q4(x),x=0..1)[1],
             Optimization[Maximize](q5(x),x=0..1)[1],
             Optimization[Maximize](q6(x),x=0..1)[1]);
ratmax := maxqj/maxq1[1];
#
# compute q1(t1)=q1(t2)=q1(maxq1)/2
plot([q1(x)],x=0..1);
t12=solve(q1(x)=maxq1[1]/2, x);
t34=solve(q1(x)=maxq1[1]*3/4,x);
#
plot([Q1(x),Q2(x),Q3(x),Q4(x),Q5(x),Q6(x)],x=0..1,
     linestyle=[SOLID,DASH,DASHDOT,DOT,DOT,DOT],
     color=[black,black,black,black,black,black],
     legend=["Q1","Q2","Q3","Q4","Q5","Q6"]);
plot([q1(x),q2(x),q3(x),q4(x),q5(x),q6(x)],x=0..1,
     linestyle=[SOLID,DASH,DASHDOT,DOT,DOT,DOT],
     color=[black,black,black,black,black,black],
     legend=["q1","q2","q3","q4","q5","q6"]);
#
dd1 := x-> w[1]*Q1(x)+Q2(x);
dd2 := x-> w[2]*Q1(x);
dd3 := x-> w[3]*Q1(x);
dd4 := x-> w[4]*Q1(x);
dd5 := x-> w[5]*Q1(x);
dd6 := x-> w[6]*Q1(x);
dd7 := x-> Q3(x);
```

```
dd8 := 0;
dd9 := 0;
dd10 := x-> Q4(x);
dd11 := x-> Q5(x);
dd12 := x-> Q6(x);
#
D0 := dd0(tau);
D1 := dd1(tau);
D2 := dd2(tau);
D3 := dd3(tau);
D4 := dd4(tau);
D5 := dd5(tau);
D6 := dd6(tau);
D7 := dd7(tau);
D8 := dd8(tau);
D9 := dd9(tau);
D10 := dd10(tau);
D11 := dd11(tau);
D12 := dd12(tau);
#
CodeGeneration[Fortran]([bb[1]=D1,bb[2]=D2,bb[3]=D3,bb[4]=D4,bb[5]=D5,bb[6]=D6,
   bb[7]=D7,bb[8]=D8,bb[9]=D9,bb[10]=D10,bb[11]=D11,bb[12]=D12],precision=double,
   output="T.f");
#
DD1 := diff(dd1(tau),tau);
DD2 := diff(dd2(tau),tau);
DD3 := diff(dd3(tau),tau);
DD4 := diff(dd4(tau),tau);
DD5 := diff(dd5(tau),tau);
DD6 := diff(dd6(tau),tau);
DD7 := diff(dd7(tau),tau);
DD8 := diff(dd8(tau),tau);
DD9 := diff(dd9(tau),tau);
```

```
DD10 := diff(dd10(tau),tau);
DD11 := diff(dd11(tau),tau);
DD12 := diff(dd12(tau),tau);
#
CodeGeneration[Fortran]([bbp[1]=DD1,bbp[2]=DD2,bbp[3]=DD3,bbp[4]=DD4,bbp[5]=DD5,
   bbp[6]=DD6,bbp[7]=DD7,bbp[8]=DD8,bbp[9]=DD9,bbp[10]=DD10,bbp[11]=DD11,
   bbp[12]=DD12],precision=double,output="F.f");
```

B.2 Source Code of Fortran for the 4/5

```
C crk45exp.f
C
C This is an experimental implementation of the order 8 explicit
C continuous RK method with defect control presented in the master
C thesis of Li Yan (2007). The parameters and calling sequence are consistent
C with those of the DVERK family of RK methods developed at Toronto
C over the last two decades. The principle focus of this version
C is the investigation of very reliable defect control schemes.
C Note that this version uses an interpolant of degree 6 and abscissa
C vector [.1, .8, .9] when invoked with INT equal 3 or 4.
C
C The particular interpolant and defect estimating strategy is
C selected by setting the global variable INT in the driver using
C labled common (common/Err/ Int). There are currently four
C possible values for this parameter (INT = 0,1,2,3,4) and these
C are discussed below and in the references cited above.
C
C Note that extensive testing has only been performed using absolute error
C control (C(1) = 1) in the call to the integrator. (The other optional
C error measures should be used with caution.) For this version the
C workspace W, that is supplied must be at least W(51,30). If problems
C with n > 51 are to be solved, W and a few temporary vectors will
```

```
C have to have their declarations adjusted appropriately.
C
C When the integrator returns every step (to provide the local
C interpolant and other information) to the calling program, (C(9) = 1),
C the trial value of y(x_i + h) is stored in the 22nd column of W,
C W(*,22) and y'_{i+1} is stored in W(*,7).
C
      subroutine CRK45 ( n, fcn, x, y, xend, tol, ind, c, nw, w )
      implicit none
      integer n, ind, nw, k,i, j, Int
      real*8 x, y(n), xend, tol, c(*), w(nw,30), temp
      real*8 aa(7,7),bt(7),cc(12),h,taus(4)
      real*8 DM, tmp2, tmp(51), tmp3(51), R(2), rr(4)
      integer  p,nstages
C Added external statement for fcn to avoid a warning message.
      external fcn
      common/Err/ Int
      common/rdtab/ aa,cc
C
C The interpolant and error control scheme is determined by the
C global flag Int. the justification and cost of each scheme is
C discussed in Hayes and Enright (2006) for Int = 0 and Int = 1;
C in Enright (2007) for Int = 2; and in this thesis for Int = 3
C and Int = 4.
C
C Int = 0 selects the standard interpolant, Int = 1 and Int = 2
C the improved and Int = 3 the new improved interpolant. The latter
C two choices required 3 extra derivative evaluations per step but
C have an easily computed asymptotically correct estimate for the
C max defect on each step. Int = 4 selects the improved interpolant
C with a validity check which involves 2 extra derivative evaluations
C on each step to verify that the defect is being reliably estimated.
C If the check fails then an additional 2 evaluations are performed
```

```
C to estimate the max defect on the step.
C
C
C
C*********************************************************************
C                                        *
C    This is a modified version of DVERK. The Changes are to use a     *
C  Continuous order 4/5 method with relaxed defect control. The 4/5    *
C  formula and interpolant were developed by Enright and Hayes (2007). *
C                                                                      *
C  This implementation  stores ytrial in W(*,22), fcn(xtrial,ytrial)   *
C  in W(*,7), and an estimate of the max defect in W(*,23). The        *
C  stages k1,k2,...k6 are stored in the first 6 columns of W.          *
C                                                                      *
C                                                                      *
C*********************************************************************
C                                                                      *
C     PURPOSE - THIS IS A RUNGE-KUTTA  SUBROUTINE  BASED  ON  VERNER'S *
C FOURTH AND FIFTH ORDER PAIR OF FORMULAS FOR FINDING APPROXIMATIONS TO *
C ?????????????????
C THE SOLUTION OF  A  SYSTEM  OF  FIRST  ORDER  ORDINARY  DIFFERENTIAL *
C EQUATIONS  WITH  INITIAL  CONDITIONS. IT ATTEMPTS TO KEEP THE GLOBAL *
C ERROR PROPORTIONAL TO  A  TOLERANCE  SPECIFIED  BY  THE  USER.  (THE *
C PROPORTIONALITY  DEPENDS  ON THE KIND OF ERROR CONTROL THAT IS USED, *
C AS WELL AS THE DIFFERENTIAL EQUATION AND THE RANGE OF INTEGRATION.)  *
C                                                                      *
C     VARIOUS OPTIONS ARE AVAILABLE TO THE USER,  INCLUDING  DIFFERENT *
C KINDS  OF  ERROR CONTROL, RESTRICTIONS ON STEP SIZES, AND INTERRUPTS *
C WHICH PERMIT THE USER TO EXAMINE THE STATE OF THE  CALCULATION  (AND *
C PERHAPS MAKE MODIFICATIONS) DURING INTERMEDIATE STAGES.              *
C                                                                      *
C     THE PROGRAM IS EFFICIENT FOR NON-STIFF SYSTEMS.  HOWEVER, A GOOD *
C VARIABLE-ORDER-ADAMS  METHOD  WILL PROBABLY BE MORE EFFICIENT IF THE *
```

```
C FUNCTION EVALUATIONS ARE VERY COSTLY.  SUCH A METHOD WOULD  ALSO  BE *
C MORE SUITABLE IF ONE WANTED TO OBTAIN A LARGE NUMBER OF INTERMEDIATE *
C SOLUTION VALUES BY INTERPOLATION, AS MIGHT BE THE CASE  FOR  EXAMPLE *
C WITH GRAPHICAL OUTPUT.                                               *
C                                                                      *
C                                    HULL-ENRIGHT-JACKSON   1/10/76    *
C                                                                      *
C***********************************************************************
C                                                                      *
C     USE - THE USER MUST SPECIFY EACH OF THE FOLLOWING                *
C                                                                      *
C     N  NUMBER OF EQUATIONS                                           *
C                                                                      *
C   FCN  NAME OF SUBROUTINE FOR EVALUATING FUNCTIONS - THE  SUBROUTINE *
C           ITSELF MUST ALSO BE PROVIDED BY THE USER - IT SHOULD BE OF *
C           THE FOLLOWING FORM                                         *
C              SUBROUTINE FCN(N, X, Y, YPRIME)                         *
C              INTEGER N                                               *
C              DOUBLE PRECISION X, Y(N), YPRIME(N)                     *
C                      *** ETC ***                                     *
C           AND IT SHOULD EVALUATE YPRIME, GIVEN N, X AND Y            *
C                                                                      *
C     X  INDEPENDENT VARIABLE - INITIAL VALUE SUPPLIED BY USER         *
C                                                                      *
C     Y  DEPENDENT VARIABLE - INITIAL VALUES OF COMPONENTS Y(1), Y(2), *
C           ..., Y(N) SUPPLIED BY USER                                 *
C                                                                      *
C  XEND  VALUE OF X TO WHICH INTEGRATION IS TO BE CARRIED OUT - IT MAY *
C           BE LESS THAN THE INITIAL VALUE OF X                        *
C                                                                      *
C   TOL  TOLERANCE - THE SUBROUTINE ATTEMPTS TO CONTROL A NORM OF  THE *
C           LOCAL  ERROR  IN  SUCH  A  WAY  THAT  THE  GLOBAL ERROR IS *
C           PROPORTIONAL TO TOL. IN SOME PROBLEMS THERE WILL BE ENOUGH *
```

```
C          DAMPING  OF  ERRORS, AS WELL AS SOME CANCELLATION, SO THAT *
C          THE GLOBAL ERROR WILL BE LESS THAN TOL. ALTERNATIVELY, THE *
C          CONTROL   CAN   BE  VIEWED  AS  ATTEMPTING  TO  PROVIDE  A *
C          CALCULATED VALUE OF Y AT XEND WHICH IS THE EXACT  SOLUTION *
C          TO  THE  PROBLEM Y' = F(X,Y) + E(X) WHERE THE NORM OF E(X) *
C          IS PROPORTIONAL TO TOL.  (THE NORM  IS  A  MAX  NORM  WITH *
C          WEIGHTS  THAT  DEPEND ON THE ERROR CONTROL STRATEGY CHOSEN *
C          BY THE USER.  THE DEFAULT WEIGHT FOR THE K-TH COMPONENT IS *
C          1/MAX(1,ABS(Y(K))),  WHICH THEREFORE PROVIDES A MIXTURE OF *
C          ABSOLUTE AND RELATIVE ERROR CONTROL.)                      *
C                                                                     *
C   IND  INDICATOR - ON INITIAL ENTRY IND MUST BE SET EQUAL TO  EITHER *
C          1  OR  2. IF THE USER DOES NOT WISH TO USE ANY OPTIONS, HE *
C          SHOULD SET IND TO 1 - ALL THAT REMAINS FOR THE USER TO  DO *
C          THEN  IS  TO  DECLARE C AND W, AND TO SPECIFY NW. THE USER *
C          MAY ALSO  SELECT  VARIOUS  OPTIONS  ON  INITIAL  ENTRY  BY *
C          SETTING IND = 2 AND INITIALIZING THE FIRST 9 COMPONENTS OF *
C          C AS DESCRIBED IN THE NEXT SECTION.  HE MAY ALSO  RE-ENTER *
C          THE  SUBROUTINE  WITH IND = 3 AS MENTIONED AGAIN BELOW. IN *
C          ANY EVENT, THE SUBROUTINE RETURNS WITH IND EQUAL TO        *
C             3 AFTER A NORMAL RETURN                                 *
C             4, 5, OR 6 AFTER AN INTERRUPT (SEE OPTIONS C(8), C(9))  *
C             -1, -2, OR -3 AFTER AN ERROR CONDITION (SEE BELOW)      *
C                                                                     *
C     C  COMMUNICATIONS VECTOR - THE DIMENSION MUST BE GREATER THAN OR *
C          EQUAL TO 24, UNLESS OPTION C(1) = 4 OR 5 IS USED, IN WHICH *
C          CASE THE DIMENSION MUST BE GREATER THAN OR EQUAL TO N+30   *
C                                                                     *
C    NW  FIRST DIMENSION OF WORKSPACE W -  MUST  BE  GREATER  THAN  OR *
C          EQUAL TO N                                                 *
C                                                                     *
C     W  WORKSPACE MATRIX - FIRST DIMENSION MUST BE NW AND SECOND MUST *
C          BE GREATER THAN OR EQUAL TO 23                             *
```

```
C                                                                      *
C     THE SUBROUTINE  WILL  NORMALLY  RETURN  WITH  IND  =  3,  HAVING *
C REPLACED THE INITIAL VALUES OF X AND Y WITH, RESPECTIVELY, THE VALUE *
C OF XEND AND AN APPROXIMATION TO Y AT XEND.  THE  SUBROUTINE  CAN  BE *
C CALLED  REPEATEDLY  WITH NEW VALUES OF XEND WITHOUT HAVING TO CHANGE *
C ANY OTHER ARGUMENT.  HOWEVER, CHANGES IN TOL, OR ANY OF THE  OPTIONS *
C DESCRIBED BELOW, MAY ALSO BE MADE ON SUCH A RE-ENTRY IF DESIRED.     *
C                                                                      *
C     THREE ERROR RETURNS ARE ALSO POSSIBLE, IN WHICH  CASE  X  AND  Y *
C WILL BE THE MOST RECENTLY ACCEPTED VALUES -                          *
C     WITH IND = -3 THE SUBROUTINE WAS UNABLE  TO  SATISFY  THE  ERROR *
C        REQUIREMENT  WITH A PARTICULAR STEP-SIZE THAT IS LESS THAN OR *
C        EQUAL TO HMIN, WHICH MAY MEAN THAT TOL IS TOO SMALL           *
C     WITH IND = -2 THE VALUE OF HMIN  IS  GREATER  THAN  HMAX,  WHICH *
C        PROBABLY  MEANS  THAT THE REQUESTED TOL (WHICH IS USED IN THE *
C        CALCULATION OF HMIN) IS TOO SMALL                             *
C     WITH IND = -1 THE ALLOWED MAXIMUM NUMBER OF FCN EVALUATIONS  HAS *
C        BEEN  EXCEEDED,  BUT  THIS  CAN ONLY OCCUR IF OPTION C(7), AS *
C        DESCRIBED IN THE NEXT SECTION, HAS BEEN USED                  *
C                                                                      *
C     THERE ARE SEVERAL CIRCUMSTANCES THAT WILL CAUSE THE CALCULATIONS *
C TO  BE  TERMINATED,  ALONG WITH OUTPUT OF INFORMATION THAT WILL HELP *
C THE USER DETERMINE THE CAUSE OF  THE  TROUBLE.  THESE  CIRCUMSTANCES *
C INVOLVE  ENTRY WITH ILLEGAL OR INCONSISTENT VALUES OF THE ARGUMENTS, *
C SUCH AS ATTEMPTING A NORMAL  RE-ENTRY  WITHOUT  FIRST  CHANGING  THE *
C VALUE OF XEND, OR ATTEMPTING TO RE-ENTER WITH IND LESS THAN ZERO.    *
C                                                                      *
C***********************************************************************
C                                                                      *
C     OPTIONS - IF THE SUBROUTINE IS ENTERED WITH IND = 1, THE FIRST 9 *
C COMPONENTS OF THE COMMUNICATIONS VECTOR ARE INITIALIZED TO ZERO, AND *
C THE SUBROUTINE USES ONLY DEFAULT VALUES  FOR  EACH  OPTION.  IF  THE *
C SUBROUTINE  IS  ENTERED  WITH IND = 2, THE USER MUST SPECIFY EACH OF *
```

```
C THESE 9  COMPONENTS - NORMALLY HE WOULD FIRST SET THEM ALL TO  ZERO, *
C AND  THEN  MAKE  NON-ZERO  THOSE  THAT  CORRESPOND TO THE PARTICULAR *
C OPTIONS HE WISHES TO SELECT. IN ANY EVENT, OPTIONS MAY BE CHANGED ON *
C RE-ENTRY  TO  THE  SUBROUTINE  -  BUT IF THE USER CHANGES ANY OF THE *
C OPTIONS, OR TOL, IN THE COURSE OF A CALCULATION HE SHOULD BE CAREFUL *
C ABOUT  HOW  SUCH CHANGES AFFECT THE SUBROUTINE - IT MAY BE BETTER TO *
C RESTART WITH IND = 1 OR 2. (COMPONENTS 10 TO 24 OF C ARE USED BY THE *
C PROGRAM  -  THE INFORMATION IS AVAILABLE TO THE USER, BUT SHOULD NOT *
C NORMALLY BE CHANGED BY HIM.)                                         *
C                                                                      *
C  C(1)  ERROR CONTROL INDICATOR - THE NORM OF THE LOCAL ERROR IS  THE *
C           MAX  NORM  OF  THE  WEIGHTED  ERROR  ESTIMATE  VECTOR, THE *
C           WEIGHTS BEING DETERMINED ACCORDING TO THE VALUE OF C(1) -  *
C              IF C(1)=1 THE WEIGHTS ARE 1 (ABSOLUTE ERROR CONTROL)    *
C              IF C(1)=2 THE WEIGHTS ARE 1/ABS(Y(K))  (RELATIVE  ERROR *
C                 CONTROL)                                             *
C              IF C(1)=3 THE  WEIGHTS  ARE  1/MAX(ABS(C(2)),ABS(Y(K))) *
C                 (RELATIVE  ERROR  CONTROL,  UNLESS ABS(Y(K)) IS LESS *
C                 THAN THE FLOOR VALUE, ABS(C(2)) )                    *
C              IF C(1)=4 THE WEIGHTS ARE 1/MAX(ABS(C(K+30)),ABS(Y(K))) *
C                 (HERE INDIVIDUAL FLOOR VALUES ARE USED)              *
C              IF C(1)=5 THE WEIGHTS ARE 1/ABS(C(K+30))                *
C              FOR ALL OTHER VALUES OF C(1), INCLUDING  C(1) = 0,  THE *
C                 DEFAULT  VALUES  OF  THE  WEIGHTS  ARE  TAKEN  TO BE *
C                 1/MAX(1,ABS(Y(K))), AS MENTIONED EARLIER             *
C           (IN THE TWO CASES C(1) = 4 OR 5 THE USER MUST DECLARE  THE *
C           DIMENSION OF C TO BE AT LEAST N+30 AND MUST INITIALIZE THE *
C           COMPONENTS C(31), C(32), ..., C(N+30).)                    *
C                                                                      *
C  C(2)  FLOOR VALUE - USED WHEN THE INDICATOR C(1) HAS THE VALUE 3    *
C                                                                      *
C  C(3)  HMIN SPECIFICATION - IF NOT ZERO, THE SUBROUTINE CHOOSES HMIN *
C           TO BE ABS(C(3)) - OTHERWISE IT USES THE DEFAULT VALUE      *
```

```
C               10*MAX(DWARF,RREB*MAX(WEIGHTED NORM Y/TOL,ABS(X))),     *
C            WHERE DWARF IS A VERY SMALL POSITIVE  MACHINE  NUMBER  AND *
C            RREB IS THE RELATIVE ROUNDOFF ERROR BOUND                  *
C                                                                       *
C  C(4)  HSTART SPECIFICATION - IF NOT ZERO, THE SUBROUTINE  WILL  USE *
C            AN  INITIAL  HMAG EQUAL TO ABS(C(4)), EXCEPT OF COURSE FOR *
C            THE RESTRICTIONS IMPOSED BY HMIN AND HMAX  -  OTHERWISE IT *
C            USES THE DEFAULT VALUE OF HMAX*(TOL)**(1/6)                *
C                                                                       *
C  C(5)  SCALE SPECIFICATION - THIS IS INTENDED TO BE A MEASURE OF THE *
C            SCALE OF THE PROBLEM - LARGER VALUES OF SCALE TEND TO MAKE *
C            THE METHOD MORE RELIABLE, FIRST  BY  POSSIBLY  RESTRICTING *
C            HMAX  (AS  DESCRIBED  BELOW) AND SECOND, BY TIGHTENING THE *
C            ACCEPTANCE REQUIREMENT - IF C(5) IS ZERO, A DEFAULT  VALUE *
C            OF  1  IS  USED.  FOR  LINEAR  HOMOGENEOUS  PROBLEMS  WITH *
C            CONSTANT COEFFICIENTS, AN APPROPRIATE VALUE FOR SCALE IS A *
C            NORM  OF  THE  ASSOCIATED  MATRIX.  FOR OTHER PROBLEMS, AN *
C            APPROXIMATION TO  AN  AVERAGE  VALUE  OF  A  NORM  OF  THE *
C            JACOBIAN ALONG THE TRAJECTORY MAY BE APPROPRIATE           *
C                                                                       *
C  C(6)  HMAX SPECIFICATION - FOUR CASES ARE POSSIBLE                   *
C            IF C(6).NE.0 AND C(5).NE.0, HMAX IS TAKEN TO BE            *
C               MIN(ABS(C(6)),2/ABS(C(5)))                              *
C            IF C(6).NE.0 AND C(5).EQ.0, HMAX IS TAKEN TO BE  ABS(C(6)) *
C            IF C(6).EQ.0 AND C(5).NE.0, HMAX IS TAKEN TO BE            *
C               2/ABS(C(5))                                             *
C            IF C(6).EQ.0 AND C(5).EQ.0, HMAX IS GIVEN A DEFAULT  VALUE *
C               OF 2                                                    *
C                                                                       *
C  C(7)  MAXIMUM NUMBER OF FUNCTION EVALUATIONS  -  IF  NOT  ZERO,  AN *
C            ERROR  RETURN WITH IND = -1 WILL BE CAUSED WHEN THE NUMBER *
C            OF FUNCTION EVALUATIONS EXCEEDS ABS(C(7))                  *
C                                                                       *
```

```
C  C(8)  INTERRUPT NUMBER  1  -  IF  NOT  ZERO,  THE  SUBROUTINE  WILL *
C           INTERRUPT   THE  CALCULATIONS  AFTER  IT  HAS  CHOSEN  ITS *
C           PRELIMINARY VALUE OF HMAG, AND JUST BEFORE CHOOSING HTRIAL *
C           AND  XTRIAL  IN  PREPARATION FOR TAKING A STEP (HTRIAL MAY *
C           DIFFER FROM HMAG IN SIGN, AND MAY  REQUIRE  ADJUSTMENT  IF *
C           XEND  IS  NEAR) - THE SUBROUTINE RETURNS WITH IND = 4, AND *
C           WILL RESUME CALCULATION AT THE POINT  OF  INTERRUPTION  IF *
C           RE-ENTERED WITH IND = 4                                    *
C                                                                      *
C  C(9)  INTERRUPT NUMBER  2  -  IF  NOT  ZERO,  THE  SUBROUTINE  WILL *
C           INTERRUPT   THE  CALCULATIONS  IMMEDIATELY  AFTER  IT  HAS *
C           DECIDED WHETHER OR NOT TO ACCEPT THE RESULT  OF  THE  MOST *
C           RECENT  TRIAL STEP, WITH IND = 5 IF IT PLANS TO ACCEPT, OR *
C           IND = 6 IF IT PLANS TO REJECT -  Y(*)  IS  THE  PREVIOUSLY *
C           ACCEPTED  RESULT, WHILE W(*,9) IS THE NEWLY COMPUTED TRIAL *
C           VALUE, AND W(*,2) IS THE UNWEIGHTED ERROR ESTIMATE VECTOR. *
C           THE  SUBROUTINE  WILL  RESUME CALCULATIONS AT THE POINT OF *
C           INTERRUPTION ON RE-ENTRY WITH IND = 5 OR 6. (THE USER  MAY *
C           CHANGE IND IN THIS CASE IF HE WISHES, FOR EXAMPLE TO FORCE *
C           ACCEPTANCE OF A STEP THAT WOULD OTHERWISE BE REJECTED,  OR *
C           VICE VERSA. HE CAN ALSO RESTART WITH IND = 1 OR 2.)        *
C                                                                      *
C The only other thing to remember is that YTRIAL is stored in         *
C the 22nd column of W (ie. in W(*,22) ) if you are using the          *
C interupt that returns every step. Otherwise the useage is the        *
C same as with DVERK.---wayne                                          *
C                                                                      *
C                                                                      *
C***********************************************************************
C                                                                      *
C  SUMMARY OF THE COMPONENTS OF THE COMMUNICATIONS VECTOR              *
C                                                                      *
C     PRESCRIBED AT THE OPTION       DETERMINED BY THE PROGRAM         *
```

```
C          OF THE USER                                                 *
C                                                                      *
C                                C(10) RREB(REL ROUNDOFF ERR BND)      *
C     C(1) ERROR CONTROL INDICATOR   C(11) DWARF (VERY SMALL MACH NO)  *
C     C(2) FLOOR VALUE               C(12) WEIGHTED NORM Y             *
C     C(3) HMIN SPECIFICATION        C(13) HMIN                        *
C     C(4) HSTART SPECIFICATION      C(14) HMAG                        *
C     C(5) SCALE SPECIFICATION       C(15) SCALE                       *
C     C(6) HMAX SPECIFICATION        C(16) HMAX                        *
C     C(7) MAX NO OF FCN EVALS       C(17) XTRIAL                      *
C     C(8) INTERRUPT NO 1            C(18) HTRIAL                      *
C     C(9) INTERRUPT NO 2            C(19) EST                         *
C                                    C(20) PREVIOUS XEND               *
C                                    C(21) FLAG FOR XEND               *
C                                    C(22) NO OF SUCCESSFUL STEPS      *
C                                    C(23) NO OF SUCCESSIVE FAILURES   *
C                                    C(24) NO OF FCN EVALS             *
C                                                                      *
C  IF C(1) = 4 OR 5, C(31), C(32), ... C(N+30) ARE FLOOR VALUES        *
C                                                                      *
C  RREB and DWARF are machine dependent constants currently set so     *
C  that they should be appropriate for most machines.  However, it may *
C  be appropriate to change them when this program is installed on a   *
C  new machine.                        K.R.J.   3 Oct 1991.            *
C                                                                      *
C***********************************************************************
C                                                                      *
C  AN OVERVIEW OF THE PROGRAM                                          *
C                                                                      *
C     BEGIN INITIALIZATION, PARAMETER CHECKING, INTERRUPT RE-ENTRIES   *
C  ......ABORT IF IND OUT OF RANGE 1 TO 6                              *
C  .     CASES - INITIAL ENTRY, NORMAL RE-ENTRY, INTERRUPT RE-ENTRIES  *
C  .     CASE 1 - INITIAL ENTRY (IND .EQ. 1 OR 2)                      *
```

```
C  V........ABORT IF N.GT.NW OR TOL.LE.0                               *
C  .        IF INITIAL ENTRY WITHOUT OPTIONS (IND .EQ. 1)              *
C  .           SET C(1) TO C(9) EQUAL TO ZERO                          *
C  .        ELSE INITIAL ENTRY WITH OPTIONS (IND .EQ. 2)               *
C  .           MAKE C(1) TO C(9) NON-NEGATIVE                          *
C  .           MAKE FLOOR VALUES NON-NEGATIVE IF THEY ARE TO BE USED   *
C  .        END IF                                                     *
C  .        INITIALIZE RREB, DWARF, PREV XEND, FLAG, COUNTS            *
C  .     CASE 2 - NORMAL RE-ENTRY (IND .EQ. 3)                         *
C  .........ABORT IF XEND REACHED, AND EITHER X CHANGED OR XEND NOT    *
C  .        RE-INITIALIZE FLAG                                         *
C  .     CASE 3 - RE-ENTRY FOLLOWING AN INTERRUPT (IND .EQ. 4 TO 6)    *
C  V        TRANSFER CONTROL TO THE APPROPRIATE RE-ENTRY POINT.......  *
C  .     END CASES                                                  .  *
C  .  END INITIALIZATION, ETC.                                      .  *
C  .                                                                V  *
C  .  LOOP THROUGH THE FOLLOWING 4 STAGES, ONCE FOR EACH TRIAL STEP .  *
C  .     STAGE 1 - PREPARE                                          .  *
C***********ERROR RETURN (WITH IND=-1) IF NO OF FCN EVALS TOO GREAT .  *
C  .        CALC SLOPE (ADDING 1 TO NO OF FCN EVALS) IF IND .NE. 6  .  *
C  .        CALC HMIN, SCALE, HMAX                                  .  *
C***********ERROR RETURN (WITH IND=-2) IF HMIN .GT. HMAX            .  *
C  .        CALC PRELIMINARY HMAG                                   .  *
C***********INTERRUPT NO 1 (WITH IND=4) IF REQUESTED.......RE-ENTRY.V  *
C  .        CALC HMAG, XTRIAL AND HTRIAL                            .  *
C  .     END STAGE 1                                                .  *
C  V     STAGE 2 - CALC YTRIAL (ADDING 7 TO NO OF FCN EVALS)        .  *
C  .     STAGE 3 - CALC THE ERROR ESTIMATE                          .  *
C  .     STAGE 4 - MAKE DECISIONS                                   .  *
C  .        SET IND=5 IF STEP ACCEPTABLE, ELSE SET IND=6            .  *
C***********INTERRUPT NO 2 IF REQUESTED....................RE-ENTRY.V  *
C  .        IF STEP ACCEPTED (IND .EQ. 5)                              *
C  .           UPDATE X, Y FROM XTRIAL, YTRIAL                         *
```

```
C  .          ADD 1 TO NO OF SUCCESSFUL STEPS                         *
C  .          SET NO OF SUCCESSIVE FAILURES TO ZERO                   *
C**************RETURN(WITH IND=3, XEND SAVED, FLAG SET) IF X .EQ. XEND *
C  .       ELSE STEP NOT ACCEPTED (IND .EQ. 6)                        *
C  .          ADD 1 TO NO OF SUCCESSIVE FAILURES                      *
C**************ERROR RETURN (WITH IND=-3) IF HMAG .LE. HMIN            *
C  .       END IF                                                     *
C  .    END STAGE 4                                                   *
C  .  END LOOP                                                        *
C  .                                                                  *
C  BEGIN ABORT ACTION                                                 *
C     OUTPUT APPROPRIATE  MESSAGE  ABOUT  STOPPING  THE  CALCULATIONS, *
C        ALONG WITH VALUES OF IND, N, NW, TOL, HMIN,  HMAX,  X,  XEND, *
C        PREVIOUS XEND,  NO OF  SUCCESSFUL  STEPS,  NO  OF  SUCCESSIVE *
C        FAILURES, NO OF FCN EVALS, AND THE COMPONENTS OF Y           *
C     STOP                                                            *
C  END ABORT ACTION                                                   *
C                                                                     *
C**********************************************************************
C        *
C**********************************************************************
C
C     *****************************************************************
C     * BEGIN INITIALIZATION, PARAMETER CHECKING, INTERRUPT RE-ENTRIES *
C     *****************************************************************
C
C initial taus for Int = 1, 2, 3, 4 respectively
      data taus / 0.23d0, 0.23d0, 0.3891d0, 0.3891d0 /
C initial 4 additional point for validation check when Int=4
      data rr / 0.2069d0, 0.5997d0, 0.2632d0, 0.5274d0 /
C different check points
C     (0.2069+0.3891)/2=0.2980; (0.5997+0.3891)/2=0.4944
C     data rr / 0.2069d0, 0.5997d0, 0.2980d0, 0.4944d0 /
```

```
C  ......ABORT IF IND OUT OF RANGE 1 TO 6
         if (ind.lt.1 .or. ind.gt.6) go to 500
C
C  CASES - INITIAL ENTRY, NORMAL RE-ENTRY, INTERRUPT RE-ENTRIES
         go to (5, 5, 45, 1111, 2222, 2222), ind
C  CASE 1 - INITIAL ENTRY (IND .EQ. 1 OR 2)
C  .........ABORT IF N.GT.NW OR TOL.LE.0
    5       if (n.gt.nw .or. tol.le.0.d0) go to 500
C
      nstages = 7
      p = 5
C     w(n,1) is yp
      call fcn(n,x,y,w(1,1) )
C
C initial Butcher tableau
      c(24) = c(24) + 1
      data cc( 1) /  0d0 /
      data cc( 2) /  0.20d0 /
      data cc( 3) /  0.30d0 /
      data cc( 4) /  0.80d0 /
      data cc( 5) /  0.888888888888888888888888888888888d0 /
      data cc( 6) /  1d0 /
      data cc( 7) /  1d0 /
      data cc( 8) /  0.86d0 /
      data cc( 9) /  0.93d0 /
      data cc(10) /  0.1d0 /
      data cc(11) /  0.8d0 /
      data cc(12) /  0.9d0 /
      data aa( 2, 1) /  0.20d0/
      data aa( 3, 1) /  0.075d0/
      data aa( 3, 2) /  0.225d0/
      data aa( 4, 1) /  0.9777777777777777777777777777777777d+00/
      data aa( 4, 2) / -3.7333333333333333333333333333333333d+00/
```

```
      data aa( 4, 3) /  3.5555555555555555555555555555555555d+00/
      data aa( 5, 1) /  2.9525986892242036274958085657674135d+00/
      data aa( 5, 2) /-11.5957933241883859167809785093735710d+00/
      data aa( 5, 3) /  9.8228928516994360615759792714525225d+00/
      data aa( 5, 4) / -0.29080932784636488340192043895747599d+00/
      data aa( 6, 1) /  2.8462752525252525252525252525252525d+00/
      data aa( 6, 2) /-10.7575757575757575757575757575757576d+00/
      data aa( 6, 3) /  8.9064227177434724604535925290642272d+00/
      data aa( 6, 4) /  0.27840909090909090909090909090909091d+00/
      data aa( 6, 5) / -0.27353130360205831903945111492281304d+000/
      data aa( 7, 1) /  0.091145833333333333333333333333333333d+00/
      data aa( 7, 2) /  0.00000000000000000000000000000000000d+00/
      data aa( 7, 3) /  0.44923629829290206648697214734950584d+00/
      data aa( 7, 4) /  0.65104166666666666666666666666666667d+00/
      data aa( 7, 5) / -0.32237617924528301886792452830188679d+00/
      data aa( 7, 6) /  0.13095238095238095238095238095238095d+00/
      data aa( 7, 7) /  0.d0 /
      do 6 j = 1,7
         bt(j) = aa(7,j)
    6 continue
C
      if (ind.eq. 2) go to 15
C  INITIAL ENTRY WITHOUT OPTIONS (IND .EQ. 1)
C  SET C(1) TO C(9) EQUAL TO 0
         do 10 k = 1, 9
            c(k) = 0.d0
   10    continue
         go to 35
   15 continue
C  INITIAL ENTRY WITH OPTIONS (IND .EQ. 2)
C  MAKE C(1) TO C(9) NON-NEGATIVE
         do 20 k = 1, 9
            c(k) = dabs(c(k))
```

```
   20    continue
C  MAKE FLOOR VALUES NON-NEGATIVE IF THEY ARE TO BE USED
         if (c(1).ne.4.d0 .and. c(1).ne.5.d0) go to 30
            do 25 k = 1, n
               c(k+30) = dabs(c(k+30))
   25       continue
   30    continue
   35 continue
C     INITIALIZE RREB, DWARF, PREV XEND, FLAG, COUNTS
      c(10) = 16.d0**(-13)
      c(11) = 1.d-28
C     SET PREVIOUS XEND INITIALLY TO INITIAL VALUE OF X
      c(20) = x
      do 40 k = 21, 24
         c(k) = 0.d0
   40 continue
      go to 50
C     CASE 2 - NORMAL RE-ENTRY (IND .EQ. 3)
C  .........ABORT IF XEND REACHED, AND EITHER X CHANGED OR XEND NOT
   45 if (c(21).ne.0.d0 .and.
     +    (x.ne.c(20) .or. xend.eq.c(20))) go to 500
C        RE-INITIALIZE FLAG
         c(21) = 0.d0
C  CASE 3 - RE-ENTRY FOLLOWING AN INTERRUPT (IND .EQ. 4 TO 6)
C     TRANSFER CONTROL TO THE APPROPRIATE RE-ENTRY POINT..........
C     THIS HAS ALREADY BEEN HANDLED BY THE COMPUTED GO TO        .
C  END CASES        V
   50    continue
C
C     END INITIALIZATION, ETC.
C
C     ******************************************************************
C     * LOOP THROUGH THE FOLLOWING 4 STAGES, ONCE FOR EACH TRIAL  STEP *
```

```
C     * UNTIL THE OCCURRENCE OF ONE OF THE FOLLOWING        *
C     *    (A) THE NORMAL RETURN (WITH IND .EQ. 3) ON REACHING XEND IN *
C     *        STAGE 4        *
C     *    (B) AN ERROR RETURN (WITH IND .LT. 0) IN STAGE 1 OR STAGE 4 *
C     *    (C) AN INTERRUPT RETURN (WITH IND  .EQ.  4,5  OR  6),  IF *
C     *        REQUESTED, IN STAGE 1 OR STAGE 4         *
C     *****************************************************************
C
99999    continue
C
C  ***************************************************************
C  * STAGE 1 - PREPARE - DO CALCULATIONS OF  HMIN,  HMAX,  ETC., *
C  * AND SOME PARAMETER  CHECKING,  AND  END  UP WITH  SUITABLE *
C  * VALUES OF HMAG, XTRIAL AND HTRIAL IN PREPARATION FOR TAKING *
C  * AN INTEGRATION STEP.         *
C  ***************************************************************
C
C***********ERROR RETURN (WITH IND=-1) IF NO OF FCN EVALS TOO GREAT
         if (c(7).eq.0.d0 .or. c(24).lt.c(7)) go to 100
            ind = -1
            return
  100    continue
C
C   Copy the f-value which has been computed previously
c
         if ((ind .ne. 5) .and. (ind .ne. 3)) go to 105
            do 102 i = 1,n
               w(i,1) = w(i,7)
  102       continue
  105    continue
C
C  CALCULATE HMIN - USE DEFAULT UNLESS VALUE PRESCRIBED
         c(13) = c(3)
```

```
         if (c(3) .ne. 0.d0) go to 165
C     CALCULATE DEFAULT VALUE OF HMIN
C     FIRST CALCULATE WEIGHTED NORM Y - C(12) - AS SPECIFIED
C     BY THE ERROR CONTROL INDICATOR C(1)
            temp = 0.d0
            if (c(1) .ne. 1.d0) go to 115
C              ABSOLUTE ERROR CONTROL - WEIGHTS ARE 1
               do 110 k = 1, n
                  temp = dmax1(temp, dabs(y(k)))
  110          continue
               c(12) = temp
               go to 160
  115       if (c(1) .ne. 2.d0) go to 120
C              RELATIVE ERROR CONTROL - WEIGHTS ARE 1/DABS(Y(K)) SO
C              WEIGHTED NORM Y IS 1
               c(12) = 1.d0
               go to 160
  120       if (c(1) .ne. 3.d0) go to 130
C              WEIGHTS ARE 1/MAX(C(2),ABS(Y(K)))
               do 125 k = 1, n
                  temp = dmax1(temp, dabs(y(k))/c(2))
  125          continue
               c(12) = dmin1(temp, 1.d0)
               go to 160
  130       if (c(1) .ne. 4.d0) go to 140
C              WEIGHTS ARE 1/MAX(C(K+30),ABS(Y(K)))
               do 135 k = 1, n
                  temp = dmax1(temp, dabs(y(k))/c(k+30))
  135          continue
               c(12) = dmin1(temp, 1.d0)
               go to 160
  140       if (c(1) .ne. 5.d0) go to 150
C              WEIGHTS ARE 1/C(K+30)
```

```
               do 145 k = 1, n
                  temp = dmax1(temp, dabs(y(k))/c(k+30))
  145          continue
               c(12) = temp
               go to 160
  150       continue
C              DEFAULT CASE - WEIGHTS ARE 1/MAX(1,ABS(Y(K)))
               do 155 k = 1, n
                  temp = dmax1(temp, dabs(y(k)))
  155          continue
               c(12) = dmin1(temp, 1.d0)
  160       continue
            c(13) = 10.d0*dmax1(c(11),c(10)*dmax1(c(12)/tol,dabs(x)))
  165    continue
C
C        CALCULATE SCALE - USE DEFAULT UNLESS VALUE PRESCRIBED
         c(15) = c(5)
         if (c(5) .eq. 0.d0) c(15) = 1.d0
C
C        CALCULATE HMAX - CONSIDER 4 CASES
C        CASE 1 BOTH HMAX AND SCALE PRESCRIBED
         if (c(6).ne.0.d0 .and. c(5).ne.0.d0)
     +      c(16) = dmin1(c(6), 2.d0/c(5))
C  CASE 2 - HMAX PRESCRIBED, BUT SCALE NOT
         if (c(6).ne.0.d0 .and. c(5).eq.0.d0) c(16) = c(6)
C  CASE 3 - HMAX NOT PRESCRIBED, BUT SCALE IS
         if (c(6).eq.0.d0 .and. c(5).ne.0.d0) c(16) = 2.d0/c(5)
C  CASE 4 - NEITHER HMAX NOR SCALE IS PROVIDED
         if (c(6).eq.0.d0 .and. c(5).eq.0.d0) c(16) = 2.d0
C
C***********ERROR RETURN (WITH IND=-2) IF HMIN .GT. HMAX
         if (c(13) .le. c(16)) go to 170
            ind = -2
```

```
            return
  170    continue
C
C  CALCULATE PRELIMINARY HMAG - CONSIDER 3 CASES
         if (ind .gt. 2) go to 175
C  CASE 1 - INITIAL ENTRY - USE PRESCRIBED VALUE OF HSTART, IF
C     ANY, ELSE DEFAULT
            c(14) = c(4)
            if (c(4) .eq. 0.d0) c(14) = c(16)*tol**(1.d0/dfloat(p))
            go to 185
  175    if (c(23) .gt. 1.d0) go to 180
C       CASE 2 - AFTER A SUCCESSFUL STEP, OR AT MOST  ONE  FAILURE,
C           USE MIN(2, .9*(TOL/EST)**(1/p))*HMAG, BUT AVOID POSSIBLE
C     OVERFLOW. THEN AVOID REDUCTION BY MORE THAN HALF.
            temp = 2.d0*c(14)
            if (tol .lt. (2.d0/.9d0)**p*c(19))
     +         temp = .9d0*(tol/c(19))**(1.d0/dfloat(p))*c(14)
            c(14) = dmax1(temp, .5d0*c(14))
            go to 185
  180    continue
C  CASE 3 - AFTER TWO OR MORE SUCCESSIVE FAILURES
            c(14) = .5d0*c(14)
  185    continue
C
C        CHECK AGAINST HMAX
         c(14) = dmin1(c(14), c(16))
C
C        CHECK AGAINST HMIN
         c(14) = dmax1(c(14), c(13))
C
C***********INTERRUPT NO 1 (WITH IND=4) IF REQUESTED
         if (c(8) .eq. 0.d0) go to 1111
            ind = 4
```

```
          return
C        RESUME HERE ON RE-ENTRY WITH IND .EQ. 4   ........RE-ENTRY..
 1111    continue
C
C  CALCULATE HMAG, XTRIAL - DEPENDING ON PRELIMINARY HMAG, XEND
         if (c(14) .ge. dabs(xend - x)) go to 190
C     DO NOT STEP MORE THAN HALF WAY TO XEND
            c(14) = dmin1(c(14), .5d0*dabs(xend - x))
            c(17) = x + dsign(c(14), xend - x)
            go to 195
  190    continue
C     HIT XEND EXACTLY
            c(14) = dabs(xend - x)
            c(17) = xend
  195    continue
C
C  CALCULATE HTRIAL
         c(18) = c(17) - x
C
C  END STAGE 1
C
C  ****************************************************************
C        STAGE 2-  Calculate ytrial using the subroutine apform
C
         h = c(18)

C        compute k_j (j=1..12) (in paper) which are stored in w_j
         call apform(n,fcn,x,y,h,w,nw,aa,bt,cc,nstages)
C
C  ADD nstages TO THE NO OF FCN EVALS
         c(24) = c(24) + dfloat(nstages - 2) + 2
         if (Int .eq. 2) c(24) = c(24)+2
         if (Int .eq. 3 .or. Int .eq. 4) c(24) = c(24)+3
```

```
C
         call defect(n,fcn,x,y,c(18),taus(Int),tmp,w(1,23),nw,w)
C
C        c(24) - No. of fcn evals
         c(24) = c(24) + 1
         temp = 0.d0
         if (c(1) .ne. 1.d0) go to 310
            do 305 k = 1, n
               temp = dmax1(temp,dabs(w(k,23)))
  305       continue
C
C At this point apply the validity check of the error estimate
C if INT has been set to 4.
C
    if (Int .eq. 4) then
               do 200 i = 1,2
                  call defect(n,fcn,x,y,c(18),rr(i),tmp,tmp3,nw,w)
                  tmp2 = 0.d0
                  do 201 k = 1,n
                     tmp2 = dmax1(tmp2, dabs(tmp3(k)))
  201             continue
                  R(i) = tmp2/temp
                  c(24) = c(24) + 1
  200          continue
C              If 1st 2 check points are around half of maximum defect,
C              we accept defect; Or we continue verify the other 2 points
               if (dmax1(dabs(R(1)-.5d0),dabs(R(2)-.5d0)).gt..2d0) then
                  DM = temp*dmax1(1.d0, dmax1(R(1),R(2)))
                  do 210 i = 3,4
                     call defect(n,fcn,x,y,c(18),rr(i),tmp,tmp3,nw,w)
                     tmp2 = 0.d0
                     do 203 k = 1,n
                        tmp2 = dmax1(tmp2, dabs(tmp3(k)))
```

```
 203               continue
                   DM = dmax1(DM, tmp2)
                   c(24) = c(24) + 1
 210             continue
                 temp = DM
               endif
             endif
             go to 360
C
C  END STAGE 2
C
C  ***************************************************************
C  * STAGE 3 - CALCULATE THE ERROR ESTIMATE EST. FIRST CALCULATE *
c        * The defect at taus and use the norm of this vector to control
C        * the stepsize. Note that w(*,2) and w(*,23) are used for temp
C        * storage.
C
C  CALCULATE THE WEIGHTED MAX NORM OF W(*,23) AS SPECIFIED BY
C  THE ERROR CONTROL INDICATOR C(1) and the global flag INT.
 310    if (c(1) .ne. 2.d0) go to 320
C     RELATIVE ERROR CONTROL
             do 315 k = 1, n
               temp = dmax1(temp, dabs(w(k,23)/y(k)))
 315         continue
             go to 360
 320    if (c(1) .ne. 3.d0) go to 330
C            WEIGHTS ARE 1/MAX(C(2),ABS(W(K,14)))
             do 325 k = 1, n
               temp = dmax1(temp, dabs(w(k,23))
    +                      / dmax1(c(2), dabs(y(k))) )
 325         continue
             go to 360
 330    if (c(1) .ne. 4.d0) go to 340
```

```
C           WEIGHTS ARE 1/MAX(C(K+30),ABS(W(K,14)))
            do 335 k = 1, n
               temp = dmax1(temp, dabs(w(k,23))
     +                     / dmax1(c(k+30), dabs(y(k))) )
  335       continue
            go to 360
  340    if (c(1) .ne. 5.d0) go to 350
C           WEIGHTS ARE 1/C(K+30)
            do 345 k = 1, n
               temp = dmax1(temp, dabs(w(k,23)/c(k+30)))
  345       continue
            go to 360
  350    continue
            do 355 k = 1, n
               temp = dmax1(temp, dabs(w(k,23))
     +                     / dmax1(1.d0, dabs(y(k))) )
  355       continue
  360    continue
C        c(19) - Est
         c(19) = temp
C
C  END STAGE 3
C
C  ***************************************************************
C  * STAGE 4 - MAKE DECISIONS.        *
C  ***************************************************************
C
C  SET IND=5 IF STEP ACCEPTABLE, ELSE SET IND=6
         ind = 5
         if (c(19) .gt. tol) ind = 6
C
C***********INTERRUPT NO 2 IF REQUESTED
         if (c(9) .eq. 0.d0) go to 2222
```

```
            return
C  RESUME HERE ON RE-ENTRY WITH IND .EQ. 5 OR 6   ...RE-ENTRY..
 2222    continue
C
         if (ind .eq. 6) go to 410
C     STEP ACCEPTED (IND .EQ. 5), SO UPDATE X, Y FROM XTRIAL,
C        YTRIAL, ADD 1 TO THE NO OF SUCCESSFUL STEPS, AND SET
C        THE NO OF SUCCESSIVE FAILURES TO ZERO
            x = c(17)
            do 400 k = 1, n
               y(k) = w(k,22)
  400       continue
            c(22) = c(22) + 1.d0
            c(23) = 0.d0
C**************RETURN(WITH IND=3, XEND SAVED, FLAG SET) IF X .EQ. XEND
            if (x .ne. xend) go to 405
               ind = 3
               c(20) = xend
               c(21) = 1.d0
               return
  405       continue
            go to 415
  410    continue
C     STEP NOT ACCEPTED (IND .EQ. 6), SO ADD 1 TO THE NO OF
C        SUCCESSIVE FAILURES
         c(23) = c(23) + 1.d0
C**************ERROR RETURN (WITH IND=-3) IF HMAG .LE. HMIN
         if (c(14) .gt. c(13)) go to 415
            ind = -3
            return
  415    continue
C
C  END STAGE 4
```

```
C
        go to 99999
C     END LOOP
C
C  BEGIN ABORT ACTION
  500 continue
C
      write(6,505) ind, tol, x, n, c(13), xend, nw, c(16), c(20),
     +      c(22), c(23), c(24), (y(k), k = 1, n)
  505 format( /// 1h0, 58hcomputation stopped in dnork with the followin
     +g values -
     +   / 1h0, 5hind =, i4, 5x, 6htol  =, 1pd13.6, 5x, 11hx         =,
     +         1pd22.15
     +   / 1h , 5hn   =, i4, 5x, 6hhmin =, 1pd13.6, 5x, 11hxend      =,
     +         1pd22.15
     +   / 1h , 5hnw  =, i4, 5x, 6hhmax =, 1pd13.6, 5x, 11hprev xend =,
     +         1pd22.15
     +   / 1h0, 14x, 27hno of successful steps    =, 0pf8.0
     +   // (1h , 1p5d24.15)                                          )
C
      stop
C
C  END ABORT ACTION
C
      end
C
C
C   This subroutine takes a single step of a RK formula
C   specified by the tableau given by a,b,cc.
C
      subroutine apform(n,fcn,x,y,h,w,nw,a,b,cc,s)
      implicit none
      integer  i,j,k,s,n,nw,Int,Intold
```

```
      real*8 x,y(n),h,w(nw,27),a(7,7),b(7),cc(12)
      external fcn, intrp
      common/Err/ Int
C
      do 100 k = 2,s
         do 90 i = 1,n
            w(i,22) = h*a(k,1)*w(i,1)
            if (k .gt. 2) then
               do 80 j = 2,k-1
                  w(i,22)=w(i,22)+h*a(k,j)*w(i,j)
 80            continue
            endif
 90      continue
         do 91 i = 1,n
C           store y(j,i+1) in w(j,22)
C           store yp(j,i+1) in w(j,7)
C           store k(j,1~10) in w(j,1~10)
            w(i,22)=w(i,22)+y(i)
 91      continue
         call fcn(n,x+cc(k)*h,w(1,22),w(1,k))
 100  continue
C
      do 120 i = 1,n
         w(i,22) = b(1)*w(i,1)
         do 110 k = 2,s
            w(i,22)=w(i,22)+b(k)*w(i,k)
 110     continue
         w(i,22) = y(i) + h*(w(i,22))
 120  continue
C
      Intold = Int
      Int = 0
      do 200 k=8,9
```

```
        call intrp(n,x,y,x+cc(k)*h,w(1,23),h,nw,w)
        call fcn(n,x+cc(k)*h,w(1,23),w(1,k))
 200  continue
      Int = Intold
C
      if (Int .eq. 2) then
         do 210 k=8,9
            call intrp(n,x,y,x+cc(k)*h,w(1,23),h,nw,w)
            call fcn(n,x+cc(k)*h,w(1,23),w(1,k))
 210     continue
      endif
C
      if (Int .eq. 3 .or. Int .eq. 4) then
         Int = 1
         do 220 k=10,12
            call intrp(n,x,y,x+cc(k)*h,w(1,23),h,nw,w)
            call fcn(n,x+cc(k)*h,w(1,23),w(1,k))
 220     continue
         Int = Intold
      endif
C
      return
      end
C
C
C  This is a relaxed defect implementation of the 4/5 rk continuous
C  method based on the pair and interpolant of Enright
C
      subroutine intrp ( n, x, y, xout, yout, htrial, nw, w)
      integer n,nw,i,Int
      real*8 x,y(n),xout,w(nw,27),htrial,yout(n)
      real*8 bt(7,4),bt2(9,6),bt3(12,6)
      real*16 qbt(7,4),qbt2(9,6)
```

```
      real* 8  polysum,tau,tau2k
      integer j,k
      common/Err/ Int
      common/indrbt/ bt,qbt,bt2,qbt2,bt3
C
      if (Int.ne.0 .and. Int.ne.2 .and. Int.ne.1 .and.
     +    Int.ne.3 .and. Int.ne.4) then
         print *, 'Int must be 0,1,2,3 or 4'
         stop
      endif
C
      if(bt(1,1) .eq. -1.D0) call intder45init
C
      tau = ( (xout) -  (x))/  (htrial)
      do 20 i = 1,n
         yout(i) = 0.d0
 20   continue
C
      if (Int .eq. 0) then
         do 10 j = 1,7
            tau2k =  (1.D0)
            polysum =  (0.d0)
            do 5 k = 1,4
               tau2k = tau2k*tau
               polysum = polysum + bt(j,k)*tau2k
 5          continue
            do 15 i = 1,n
               yout(i)= yout(i)+ polysum * w(i,j)
 15         continue
 10      continue
         do 14 i = 1,n
            yout(i)=  y(i)+ htrial* yout(i)
 14      continue
```

```
      endif
C
      if (Int .eq. 1 .or. Int .eq. 2) then
         do 16 j = 1,9
            tau2k =  (1.D0)
            polysum =  (0.d0)
            do 18 k = 1,6
               polysum = polysum + bt2(j,k)*tau2k
               tau2k = tau2k*tau
 18          continue
            do 17 i = 1,n
               yout(i)= yout(i) + polysum * w(i,j)
 17         continue
 16      continue
         do 19 i = 1,n
            yout(i)=  y(i)+ htrial* yout(i)
 19      continue
      endif
C
      if (Int .eq. 3 .or. Int .eq. 4) then
         do 22 j = 1,12
            tau2k = (1.D0)
            polysum = (0.D0)
            do 24 k = 1,6
               tau2k = tau2k*tau
               polysum = polysum + bt3(j,k)*tau2k
 24         continue
            do 26 i = 1,n
               yout(i) = yout(i) + polysum*w(i,j)
 26         continue
 22      continue
         do 27 i = 1,n
            yout(i) = y(i) + htrial*yout(i)
```

```
 27      continue
      endif
C
      return
      end
C
C
      subroutine deriv ( n, x, y, xout, ypout, htrial, nw, w)
C
      integer n,nw,i,j,k,Int
      real*8 x,y(n),xout,w(nw,27),htrial,ypout(n)
      real*8 bt(7,4),bt2(9,6),bt3(12,6)
      real*16 qbt(7,4),qbt2(9,6)
      real* 8  polysum,tau,tau2k
      common/Err/ Int
      common /indrbt/bt,qbt,bt2,qbt2,bt3
C
      if(bt(1,1) .eq. -1.D0) call intder45init
C
      tau = ( (xout) -  (x))/  (htrial)
      do 20 i = 1,n
         ypout(i) = 0.d0
   20 continue
C
      if (Int .eq. 0) then
         do 2 j = 1,7
            tau2k = (1.D0)
            polysum = (0d0)
            do 3 k = 1,4
               polysum = polysum+k*bt(j,k)*tau2k
               tau2k = tau*tau2k
    3       continue
            do 4 i =1,n
```

```
               ypout(i)=ypout(i)+polysum*w(i,j)
   4        continue
   2     continue
      endif
C
      if (Int .eq. 1 .or. Int .eq. 2) then
         do 10 j = 1,9
            tau2k = (1.d0)
            polysum =  (0.d0)
            do 5 k = 1,6
               if (k .gt. 1) then
                  polysum=polysum+(k-1)*bt2(j,k)*tau2k
                  tau2k = tau*tau2k
               endif
    5       continue
            do 7 i = 1,n
               ypout(i)=ypout(i)+polysum*w(i,j)
    7       continue
   10    continue
      endif
C
      if (Int .eq. 3 .or. Int .eq. 4) then
         do 22 j = 1,12
            tau2k = (1.d0)
            polysum = (0.d0)
            do 25 k = 1,6
               polysum = polysum + k*bt3(j,k)*tau2k
               tau2k = tau2k*tau
  25        continue
            do 27 i = 1,n
               ypout(i) = ypout(i) + polysum*w(i,j)
  27        continue
  22     continue
```

```
      endif
C
      return
      end
C
C
C Compute the defect at x+h*tau.  Use precomputed
C polynomial co-efficients if tau=taustar, for better accuracy.
C Future work: make it accurate with arbitrary tau.
C
      subroutine defect (n,f,x,y,h,tau,ypout,defout,nw,w)
      integer i,n,nw
      external f
      real*8 x,y(n),h,tau,ypout(n),defout(n),w(nw,27)
C
      call intrp (n,x,y,x+tau*h,ypout,h,nw,w)
      call f(n,x+tau*h,ypout,defout)
      call deriv (n,x,y,x+tau*h,ypout,h,nw,w)
C
      do 1600 i = 1,n
        defout(i)=(defout(i)-ypout(i))
1600  continue
C
      end
C
C
C Put this inside 1  because we only want to define this subroutine
C once, even though it initialized both the quad and double tables.
C
      subroutine intder45init
      integer j,k
      real*8 bt(7,4),bt2(9,6),bt3(12,6)
      real*16 qbt(7,4),qbt2(9,6)
```

```
      common /indrbt/bt,qbt,bt2,qbt2,bt3
C
C The value -1 indicates the tables are currently uninitialized.
      data bt (1,1) /  -1.D0/
C The polynomial co-efficients for the standard interpolant
      qbt( 1, 1) =  1q0
      qbt( 1, 2) = -183q0/64q0
      qbt( 1, 3) =  37q0/12q0
      qbt( 1, 4) = -145q0/128q0
      qbt( 2, 1) =  0q0
      qbt( 2, 2) =  0q0
      qbt( 2, 3) =  0q0
      qbt( 2, 4) =  0q0
      qbt( 3, 1) =  0q0
      qbt( 3, 2) =  1500q0/371q0
      qbt( 3, 3) = -1000q0/159q0
      qbt( 3, 4) =  1000q0/371q0
      qbt( 4, 1) =  0q0
      qbt( 4, 2) = -125q0/32q0
      qbt( 4, 3) =  125q0/12q0
      qbt( 4, 4) = -375q0/64q0
      qbt( 5, 1) =  0q0
      qbt( 5, 2) =  9477q0/3392q0
      qbt( 5, 3) = -729q0/106q0
      qbt( 5, 4) =  25515q0/6784q0
      qbt( 6, 1) =  0q0
      qbt( 6, 2) = -11q0/7q0
      qbt( 6, 3) =  11q0/3q0
      qbt( 6, 4) = -55q0/28q0
      qbt( 7, 1) =  0q0
      qbt( 7, 2) =  3q0/2q0
      qbt( 7, 3) = -4q0
      qbt( 7, 4) =  5q0/2q0
```

```
C The polynomial co-efficients for the improved interpolant

      qbt2(1,1) = 0q0

      qbt2(1,2) = 1q0

      qbt2(1,3) = -1708582621q0/524156928q0

      qbt2(1,4) = 1232939669q0/262078464q0

      qbt2(1,5) = -1663764925q0/524156928q0

      qbt2(1,6) = 208375q0/253952q0

      qbt2(2,1) = 0q0

      qbt2(2,2) = 0q0

      qbt2(2,3) = 0q0

      qbt2(2,4) = 0q0

      qbt2(2,5) = 0q0

      qbt2(2,6) = 0q0

      qbt2(3,1) = 0q0

      qbt2(3,2) = 0q0

      qbt2(3,3) = 499875q0/94976q0

      qbt2(3,4) = -1618625q0/142464q0

      qbt2(3,5) = 871875q0/94976q0

      qbt2(3,6) = -15625q0/5936q0

      qbt2(4,1) = 0q0

      qbt2(4,2) = 0q0

      qbt2(4,3) = 499875q0/65536q0

      qbt2(4,4) = -1618625q0/98304q0

      qbt2(4,5) = 871875q0/65536q0

      qbt2(4,6) = -15625q0/4096q0

      qbt2(5,1) = 0q0

      qbt2(5,2) = 0q0

      qbt2(5,3) = -26237439q0/6946816q0

      qbt2(5,4) = 28319463q0/3473408q0

      qbt2(5,5) = -45762975q0/6946816q0

      qbt2(5,6) = 820125q0/434176q0

      qbt2(6,1) = 0q0

      qbt2(6,2) = 0q0
```

```
      qbt2(6,3) = 43989q0/28672q0
      qbt2(6,4) = -142439q0/43008q0
      qbt2(6,5) = 76725q0/28672q0
      qbt2(6,6) = -1375q0/1792q0
      qbt2(7,1) = 0q0
      qbt2(7,2) = 0q0
      qbt2(7,3) = -2291427q0/100352q0
      qbt2(7,4) = 3838251q0/50176q0
      qbt2(7,5) = -8579075q0/100352q0
      qbt2(7,6) = 199625q0/6272q0
      qbt2(8,1) = 0q0
      qbt2(8,2) = 0q0
      qbt2(8,3) = -47953125q0/1078784q0
      qbt2(8,4) = 74828125q0/539392q0
      qbt2(8,5) = -155453125q0/1078784q0
      qbt2(8,6) = 78125q0/1568q0
      qbt2(9,1) = 0q0
      qbt2(9,2) = 0q0
      qbt2(9,3) = 8734375q0/145824q0
      qbt2(9,4) = -14359375q0/72912q0
      qbt2(9,5) = 31234375q0/145824q0
      qbt2(9,6) = -234375q0/3038q0
C The polynomial co-efficients for the new improved interpolant
      bt3(1,1) = 1.d0
      bt3(1,2) = -0.13303D5 / 0.1584D4
      bt3(1,3) = 0.791347D6 / 0.28512D5
      bt3(1,4) = - 0.1589515D7 / 0.38016D5
      bt3(1,5) = 0.35045D5 / 0.1188D4
      bt3(1,6) = - 0.113375D6 / 0.14256D5
      bt3(2,1) = 0.0D0
      bt3(2,2) = 0.0D0
      bt3(2,3) = 0.0D0
      bt3(2,4) = 0.0D0
```

```
bt3(2,5) = 0.0D0
bt3(2,6) = 0.0D0
bt3(3,1) = 0.0D0
bt3(3,2) = -0.12000D5 / 0.4081D4
bt3(3,3) = 0.962000D6 / 0.36729D5
bt3(3,4) = - 0.672500D6 / 0.12243D5
bt3(3,5) = 0.80000D5 / 0.1749D4
bt3(3,6) = - 0.500000D6 / 0.36729D5
bt3(4,1) = 0.0D0
bt3(4,2) = -0.375D3 / 0.88D2
bt3(4,3) = 0.60125D5 / 0.1584D4
bt3(4,4) = - 0.168125D6 / 0.2112D4
bt3(4,5) = 0.4375D4 / 0.66D2
bt3(4,6) = - 0.15625D5 / 0.792D3
bt3(5,1) = 0.0D0
bt3(5,2) = 0.19683D5 / 0.9328D4
bt3(5,3) = - 0.350649D6 / 0.18656D5
bt3(5,4) = 0.2941515D7 / 0.74624D5
bt3(5,5) = - 0.76545D5 / 0.2332D4
bt3(5,6) = 0.91125D5 / 0.9328D4
bt3(6,1) = 0.0D0
bt3(6,2) = -0.6D1 / 0.7D1
bt3(6,3) = 0.481D3 / 0.63D2
bt3(6,4) = -0.1345D4 / 0.84D2
bt3(6,5) = 0.40D2 / 0.3D1
bt3(6,6) = - 0.250D3 / 0.63D2
bt3(7,1) = 0.0D0
bt3(7,2) = 0.62D2 / 0.33D2
bt3(7,3) = - 0.16099D5 / 0.891D3
bt3(7,4) = 0.14095D5 / 0.297D3
bt3(7,5) = - 0.14620D5 / 0.297D3
bt3(7,6) = 0.16000D5 / 0.891D3
bt3(8,1) = 0.0D0
```

```
      bt3(8,2) = 0.0D0
      bt3(8,3) = 0.0D0
      bt3(8,4) = 0.0D0
      bt3(8,5) = 0.0D0
      bt3(8,6) = 0.0D0
      bt3(9,1) = 0.0D0
      bt3(9,2) = 0.0D0
      bt3(9,3) = 0.0D0
      bt3(9,4) = 0.0D0
      bt3(9,5) = 0.0D0
      bt3(9,6) = 0.0D0
      bt3(10,1) = 0.0D0
      bt3(10,2) = 0.2500D4 / 0.231D3
      bt3(10,3) = - 0.304250D6 / 0.6237D4
      bt3(10,4) = 0.170750D6 / 0.2079D4
      bt3(10,5) = - 0.127250D6 / 0.2079D4
      bt3(10,6) = 0.106250D6 / 0.6237D4
      bt3(11,1) = 0.0D0
      bt3(11,2) = 0.375D3 / 0.56D2
      bt3(11,3) = - 0.15875D5 / 0.252D3
      bt3(11,4) = 0.26125D5 / 0.168D3
      bt3(11,5) = - 0.3125D4 / 0.21D2
      bt3(11,6) = 0.3125D4 / 0.63D2
      bt3(12,1) = 0.0D0
      bt3(12,2) = -0.500D3 / 0.99D2
      bt3(12,3) = 0.43750D5 / 0.891D3
      bt3(12,4) = - 0.39250D5 / 0.297D3
      bt3(12,5) = 0.40750D5 / 0.297D3
      bt3(12,6) = - 0.43750D5 / 0.891D3
C
C Now copy the values from quad to double
      do 1 j=1,9
      do 1 k=1,6
```

```
      if(j.le.7.and.k.le.4) then
         bt(j,k)=qbt(j,k)
      end if
         bt2(j,k)=qbt2(j,k)
 1  continue
    end
```

Appendix C

Numerical Results for the 4/5

C.1 Reliability of the Method

NONSTIFF DETEST PACKAGE   OPTION= 5, NORMEF= 3, NRMTYP= 1, GLBDEF= 0   ON   Sun 4/280-32

GROUP 1   CRK45   INT=a3

A1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.004	109	8	0.01	0.24	0.971	0.000	0.000
-2.00	0.000	-0.004	109	9	0.01	0.18	0.721	0.000	0.000
-3.00	0.000	-0.005	133	11	0.01	0.12	0.608	0.000	0.000
-4.00	0.000	-0.007	169	14	0.12	0.12	0.615	0.000	0.000
-5.00	0.000	-0.009	241	20	0.00	0.14	0.666	0.000	0.000
-6.00	0.000	-0.013	337	28	0.00	0.16	0.575	0.000	0.000
-7.00	0.000	-0.019	493	41	0.01	0.17	0.606	0.000	0.000
-8.00	0.000	-0.028	721	60	0.01	0.19	0.614	0.000	0.000
-9.00	0.000	-0.043	1093	91	0.02	0.20	0.588	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.537E-02 *(TOL** 1.002) APPROX,   R.M.S. RESIDUAL= 3.3E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.158 *(TOL** 0.996) APPROX,   R.M.S. RESIDUAL= 9.1E-02 OVER 9 VALUES

MAXIMUM DEFECT = 0.809 *(TOL** 1.018) APPROX,   R.M.S. RESIDUAL= 4.7E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS

10** -2	-1.87	0.000	-0.004	109	8
10** -3	-2.86	0.000	-0.005	129	10
10** -4	-3.84	0.000	-0.006	163	13
10** -5	-4.82	0.000	-0.009	227	18
10** -6	-5.80	0.000	-0.012	317	26
10** -7	-6.78	0.000	-0.018	459	38
10** -8	-7.76	0.000	-0.026	667	55
10** -9	-8.75	0.000	-0.039	998	83

A2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.003	73	6	0.01	0.14	0.603	0.000	0.000
-2.00	0.000	-0.003	85	7	0.02	0.11	0.616	0.000	0.000
-3.00	0.000	-0.004	97	8	0.12	0.12	0.561	0.000	0.000
-4.00	0.000	-0.005	133	11	0.09	0.16	0.814	0.000	0.000
-5.00	0.000	-0.007	181	14	0.21	0.27	0.518	0.000	0.000
-6.00	0.000	-0.008	229	19	0.29	0.42	0.725	0.000	0.000
-7.00	0.000	-0.011	313	26	0.46	0.61	0.929	0.000	0.000
-8.00	0.000	-0.016	433	36	0.66	0.81	0.640	0.000	0.000
-9.00	0.000	-0.022	613	51	0.91	1.00	0.758	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.014E-02 *(TOL** 0.766) APPROX, R.M.S. RESIDUAL= 1.7E-01 OVER 9 VALUES

```
MAXIMUM  GLOBAL ERROR= 6.568E-02 *(TOL** 0.870) APPROX,     R.M.S. RESIDUAL= 9.9E-02 OVER  9 VALUES
MAXIMUM DEFECT       = 0.574     *(TOL** 0.986) APPROX,     R.M.S. RESIDUAL= 6.8E-02 OVER  9 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.78	0.000	-0.003	82	6
10** -3	-2.80	0.000	-0.003	94	7
10** -4	-3.81	0.000	-0.005	126	10
10** -5	-4.83	0.000	-0.006	172	13
10** -6	-5.84	0.000	-0.008	221	18
10** -7	-6.85	0.000	-0.011	300	24
10** -8	-7.87	0.000	-0.015	417	34
10** -9	-8.88	0.000	-0.022	591	49

A3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.009	181	10	2.25	2.24	0.862	0.000	0.000
-2.00	0.000	-0.010	205	11	2.79	2.83	1.010	0.091	0.000
-3.00	0.000	-0.014	289	18	4.83	4.82	0.992	0.000	0.000
-4.00	0.000	-0.020	409	26	8.20	8.12	0.996	0.000	0.000
-5.00	0.000	-0.029	577	39	4.67	4.59	0.952	0.000	0.000
-6.00	0.000	-0.044	889	59	5.20	5.17	0.904	0.000	0.000

-7.00	0.000	-0.067	1357	88	5.58	5.57	1.203	0.045	0.000
-8.00	0.000	-0.095	1921	131	4.16	4.16	1.044	0.008	0.000
-9.00	0.000	-0.137	2773	197	3.10	3.09	1.694	0.051	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.49 *(TOL** 0.983) APPROX, R.M.S. RESIDUAL= 1.6E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 3.49 *(TOL** 0.984) APPROX, R.M.S. RESIDUAL= 1.5E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.813 *(TOL** 0.978) APPROX, R.M.S. RESIDUAL= 5.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.93	0.000	-0.009	179	9
10** -2	-1.95	0.000	-0.010	203	10
10** -3	-2.98	0.000	-0.014	287	17
10** -4	-4.00	0.000	-0.020	408	25
10** -5	-5.02	0.000	-0.029	583	39
10** -6	-6.05	0.000	-0.045	910	60
10** -7	-7.07	0.000	-0.069	1395	90
10** -8	-8.09	0.000	-0.099	1998	137

A4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.01	0.01	0.004	0.000	0.000

-2.00	0.000	-0.002	49	4	0.06	0.12	0.040	0.000	0.000
-3.00	0.000	-0.002	49	4	0.57	1.23	0.398	0.000	0.000
-4.00	0.000	-0.004	73	5	0.73	1.48	0.499	0.000	0.000
-5.00	0.000	-0.004	85	6	2.19	4.17	0.810	0.000	0.000
-6.00	0.000	-0.005	97	8	2.82	4.92	0.917	0.000	0.000
-7.00	0.000	-0.007	145	12	2.86	5.58	0.803	0.000	0.000
-8.00	0.000	-0.011	217	17	1.93	5.38	0.698	0.000	0.000
-9.00	0.000	-0.018	361	27	1.23	5.34	1.051	0.074	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.427E-02 *(TOL** 0.751) APPROX, R.M.S. RESIDUAL= 4.8E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 4.628E-02 *(TOL** 0.712) APPROX, R.M.S. RESIDUAL= 4.6E-01 OVER 9 VALUES

MAXIMUM DEFECT = 2.797E-02 *(TOL** 0.788) APPROX, R.M.S. RESIDUAL= 3.9E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.83	0.000	-0.002	49	4
10** -4	-3.10	0.000	-0.003	51	4
10** -5	-4.37	0.000	-0.004	77	5
10** -6	-5.64	0.000	-0.005	92	7
10** -7	-6.91	0.000	-0.007	140	11
10** -8	-8.18	0.000	-0.012	242	18

A5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.01	0.01	0.001	0.000	0.000
-2.00	0.000	-0.002	49	4	0.07	0.07	0.005	0.000	0.000
-3.00	0.000	-0.002	49	4	0.72	0.72	0.051	0.000	0.000
-4.00	0.000	-0.002	61	5	2.66	2.67	0.442	0.000	0.000
-5.00	0.000	-0.003	85	7	4.08	4.22	0.728	0.000	0.000
-6.00	0.000	-0.005	121	10	5.82	5.94	0.855	0.000	0.000
-7.00	0.000	-0.007	169	14	7.25	7.54	0.705	0.000	0.000
-8.00	0.000	-0.010	241	20	8.97	9.25	0.892	0.000	0.000
-9.00	0.000	-0.015	361	30	9.43	10.12	0.776	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 2.645E-02 *(TOL** 0.656) APPROX, R.M.S. RESIDUAL= 4.6E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 2.661E-02 *(TOL** 0.654) APPROX, R.M.S. RESIDUAL= 4.5E-01 OVER 9 VALUES

MAXIMUM DEFECT = 9.608E-03 *(TOL** 0.734) APPROX, R.M.S. RESIDUAL= 3.9E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.34	0.000	-0.002	49	4
10** -4	-2.70	0.000	-0.002	49	4
10** -5	-4.07	0.000	-0.003	62	5
10** -6	-5.43	0.000	-0.004	100	8
10** -7	-6.79	0.000	-0.006	159	13

10** -8	-8.16	0.000	-0.010	259	21

SUMMARY OVER GROUP 1

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.020	461	32	2.25	2.24	0.971	0.000	0.000
-2.00	0.000	-0.022	497	35	2.79	2.83	1.010	0.029	0.000
-3.00	0.000	-0.027	617	45	4.83	4.82	0.992	0.000	0.000
-4.00	0.000	-0.038	845	61	8.20	8.12	0.996	0.000	0.000
-5.00	0.000	-0.052	1169	86	4.67	4.59	0.952	0.000	0.000
-6.00	0.000	-0.075	1673	124	5.82	5.94	0.917	0.000	0.000
-7.00	0.000	-0.112	2477	181	7.25	7.54	1.203	0.022	0.000
-8.00	0.000	-0.159	3533	264	8.97	9.25	1.044	0.004	0.000
-9.00	0.000	-0.235	5201	396	9.43	10.12	1.694	0.030	0.000

GROUP 2 CRK45 INT=a3

B1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	**************		2257	100	********	113370.52	54.051	0.130	0.030
	METHOD FAILED AT X = 1.63639E+01								
-2.00	0.000	-0.031	661	38	4.83	16.93	0.874	0.000	0.000

```
 -3.00    0.000 -0.040      853      50      9.78     40.81     0.924     0.000      0.000
 -4.00    0.000 -0.056     1213      70      5.88     24.08     0.782     0.000      0.000
 -5.00    0.000 -0.075     1609      99      8.67     38.86     0.926     0.000      0.000
 -6.00    0.000 -0.101     2161     144     11.67     51.60     0.990     0.000      0.000
 -7.00    0.000 -0.132     2845     211     11.74     51.25     0.997     0.000      0.000
 -8.00    0.000 -0.185     3973     314     13.93     59.57     0.973     0.000      0.000
 -9.00    0.000 -0.272     5845     478     15.44      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
            TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  3.90652E-03, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
           SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
           ENDPOINT GLOBAL ERROR=  4.29     *(TOL** 0.937) APPROX,      R.M.S. RESIDUAL= 8.0E-02 OVER  8 VALUES
           MAXIMUM  GLOBAL ERROR=  15.3     *(TOL** 0.923) APPROX,      R.M.S. RESIDUAL= 9.9E-02 OVER  7 VALUES
           MAXIMUM DEFECT       = 0.811     *(TOL** 0.989) APPROX,      R.M.S. RESIDUAL= 2.7E-02 OVER  7 VALUES
NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
ACCURACY LOG10 TOL                 CALLS   STEPS
10** -2   -1.93    0.000 -0.030      647      37
10** -3   -2.94    0.000 -0.039      841      49
10** -4   -3.95    0.000 -0.056     1195      69
10** -5   -4.96    0.000 -0.074     1594      97
10** -6   -5.97    0.000 -0.100     2147     142
10** -7   -6.99    0.000 -0.132     2835     210
10** -8   -8.00    0.000 -0.185     3969     313
```

B2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.009	253	20	0.01	0.07	0.852	0.000	0.000	
-2.00	0.000	-0.010	277	22	0.00	0.07	0.870	0.000	0.000	
-3.00	0.000	-0.012	325	25	0.00	0.08	1.024	0.040	0.000	
-4.00	0.000	-0.013	361	29	0.00	0.08	0.978	0.000	0.000	
-5.00	0.000	-0.016	445	35	0.01	0.11	0.902	0.000	0.000	
-6.00	0.000	-0.021	577	45	0.01	0.13	0.657	0.000	0.000	
-7.00	0.000	-0.028	757	62	0.00	0.15	0.609	0.000	0.000	
-8.00	0.000	-0.040	1093	91	0.01	0.17	0.612	0.000	0.000	
-9.00	0.000	-0.061	1657	138	0.02	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 2.58297E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.053E-02 *(TOL** 0.994) APPROX, R.M.S. RESIDUAL= 7.3E-02 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 5.549E-02 *(TOL** 0.942) APPROX, R.M.S. RESIDUAL= 3.1E-02 OVER 8 VALUES

MAXIMUM DEFECT = 1.07 *(TOL** 1.028) APPROX, R.M.S. RESIDUAL= 5.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-1.00	0.000	-0.009	253	20
10** -2	-1.97	0.000	-0.010	276	21
10** -3	-2.95	0.000	-0.012	322	24

10** -4	-3.92	0.000	-0.013	358	28
10** -5	-4.89	0.000	-0.016	435	34
10** -6	-5.86	0.000	-0.021	558	43
10** -7	-6.84	0.000	-0.027	727	59
10** -8	-7.81	0.000	-0.038	1028	85

B3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.004	121	9	0.01	0.20	0.696	0.000	0.000
-2.00	0.000	-0.005	145	11	0.02	0.09	0.410	0.000	0.000
-3.00	0.000	-0.006	193	14	0.01	0.10	0.532	0.000	0.000
-4.00	0.000	-0.008	229	18	0.01	0.12	0.792	0.000	0.000
-5.00	0.000	-0.011	337	25	0.03	0.12	0.631	0.000	0.000
-6.00	0.000	-0.015	445	34	0.09	0.13	0.802	0.000	0.000
-7.00	0.000	-0.021	625	49	0.12	0.15	0.962	0.000	0.000
-8.00	0.000	-0.029	877	72	0.34	0.34	0.875	0.000	0.000
-9.00	0.000	-0.046	1381	115	0.54	0.54	0.626	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.888E-03 *(TOL** 0.779) APPROX, R.M.S. RESIDUAL= 2.3E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 7.902E-02 *(TOL** 0.936) APPROX, R.M.S. RESIDUAL= 1.8E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.530 *(TOL** 0.978) APPROX, R.M.S. RESIDUAL= 9.3E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.76	0.000	-0.005	139	10
10** -3	-2.79	0.000	-0.006	182	13
10** -4	-3.81	0.000	-0.007	222	17
10** -5	-4.83	0.000	-0.011	318	23
10** -6	-5.85	0.000	-0.014	429	32
10** -7	-6.88	0.000	-0.020	602	47
10** -8	-7.90	0.000	-0.028	851	69
10** -9	-8.92	0.000	-0.045	1340	111

B4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.038	373	20	6.48	6.44	0.967	0.000	0.000
-2.00	0.000	-0.040	385	24	4.11	4.07	0.806	0.000	0.000
-3.00	0.000	-0.057	553	35	1.85	1.71	0.996	0.000	0.000
-4.00	0.000	-0.072	697	48	1.50	1.94	0.976	0.000	0.000
-5.00	0.000	-0.100	973	71	2.81	3.68	0.981	0.000	0.000
-6.00	0.000	-0.143	1393	105	1.06	2.16	0.872	0.000	0.000
-7.00	0.000	-0.203	1981	155	1.62	2.44	0.981	0.000	0.000
-8.00	0.000	-0.298	2905	232	1.99	2.46	0.954	0.000	0.000
-9.00	0.000	-0.440	4285	354	2.56	2.69	0.934	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.96 *(TOL** 1.047) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 3.99 *(TOL** 1.030) APPROX, R.M.S. RESIDUAL= 1.5E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.922 *(TOL** 0.998) APPROX, R.M.S. RESIDUAL= 2.8E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.038	372	19
10** -2	-1.97	0.000	-0.039	384	23
10** -3	-2.97	0.000	-0.056	547	34
10** -4	-3.97	0.000	-0.071	692	47
10** -5	-4.97	0.000	-0.099	965	70
10** -6	-5.97	0.000	-0.142	1382	104
10** -7	-6.98	0.000	-0.202	1966	153
10** -8	-7.98	0.000	-0.296	2884	230
10** -9	-8.98	0.000	-0.437	4256	351

B5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.015	313	16	1.25	1.30	0.495	0.000	0.000
-2.00	0.000	-0.017	373	22	0.53	1.19	0.959	0.000	0.000
-3.00	0.000	-0.022	481	29	0.43	0.41	0.948	0.000	0.000

-4.00	0.000	-0.029	625	39	3.09	3.98	1.131	0.051	0.000
-5.00	0.000	-0.042	913	56	0.90	1.38	1.540	0.018	0.000
-6.00	0.000	-0.056	1213	80	1.85	2.32	1.436	0.062	0.000
-7.00	0.000	-0.078	1693	117	1.87	2.48	1.099	0.017	0.000
-8.00	0.000	-0.107	2305	176	4.34	5.94	0.998	0.000	0.000
-9.00	0.000	-0.156	3361	276	8.73	11.84	1.016	0.004	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 0.422 *(TOL** 0.880) APPROX, R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.554 *(TOL** 0.879) APPROX, R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.764 *(TOL** 0.974) APPROX, R.M.S. RESIDUAL= 1.1E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.91	0.000	-0.014	307	15
10** -2	-1.93	0.000	-0.017	368	21
10** -3	-2.96	0.000	-0.022	476	28
10** -4	-3.99	0.000	-0.029	622	38
10** -5	-5.01	0.000	-0.042	916	56
10** -6	-6.04	0.000	-0.057	1231	81
10** -7	-7.06	0.000	-0.080	1732	120
10** -8	-8.09	0.000	-0.111	2400	184

SUMMARY OVER GROUP 2

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.066	1060	65	6.48	6.44	0.967	0.000	0.000
-2.00	0.000	-0.103	1841	117	4.83	16.93	0.959	0.000	0.000
-3.00	0.000	-0.137	2405	153	9.78	40.81	1.024	0.007	0.000
-4.00	0.000	-0.178	3125	204	5.88	24.08	1.131	0.010	0.000
-5.00	0.000	-0.245	4277	286	8.67	38.86	1.540	0.003	0.000
-6.00	0.000	-0.336	5789	408	11.67	51.60	1.436	0.012	0.000
-7.00	0.000	-0.463	7901	594	11.74	51.25	1.099	0.003	0.000
-8.00	0.000	-0.659	11153	885	13.93	59.57	0.998	0.000	0.000
-9.00	0.000	-0.975	16529	1361	15.44	11.84	1.016	0.001	0.000

(LOC ASSESS ON 745)

GROUP 3 CRK45 INT=a3

C1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.008	169	13	0.00	0.10	0.657	0.000	0.000
-2.00	0.000	-0.009	193	15	0.00	0.15	0.601	0.000	0.000
-3.00	0.000	-0.010	217	18	0.05	0.27	0.726	0.000	0.000
-4.00	0.000	-0.014	301	25	0.12	0.36	0.724	0.000	0.000
-5.00	0.000	-0.020	445	37	0.08	0.48	0.617	0.000	0.000

-6.00	0.000	-0.030	649	54	0.09	0.55	0.632	0.000	0.000
-7.00	0.000	-0.045	985	82	0.11	0.59	0.588	0.000	0.000
-8.00	0.000	-0.068	1501	125	0.18	0.62	0.600	0.000	0.000
-9.00	0.000	-0.106	2329	194	0.26	0.64	0.602	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 9.257E-03 *(TOL** 0.831) APPROX, R.M.S. RESIDUAL= 2.0E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.115 *(TOL** 0.902) APPROX, R.M.S. RESIDUAL= 1.0E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.687 *(TOL** 1.007) APPROX, R.M.S. RESIDUAL= 2.8E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.82	0.000	-0.009	188	14
10** -3	-2.82	0.000	-0.010	212	17
10** -4	-3.81	0.000	-0.013	285	23
10** -5	-4.81	0.000	-0.019	416	34
10** -6	-5.80	0.000	-0.028	607	50
10** -7	-6.79	0.000	-0.042	915	76
10** -8	-7.79	0.000	-0.063	1390	115
10** -9	-8.78	0.000	-0.098	2145	178

C2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV

```
 -1.00    0.000 -0.057      709      57      0.00      0.02     0.643     0.000      0.000
 -2.00    0.000 -0.056      697      58      0.00      0.04     0.829     0.000      0.000
 -3.00    0.000 -0.059      733      60      0.01      0.10     0.850     0.000      0.000
 -4.00    0.000 -0.064      805      65      0.00      0.15     0.982     0.000      0.000
 -5.00    0.000 -0.074      925      73      0.01      0.16     0.994     0.000      0.000
 -6.00    0.000 -0.090     1129      91      0.00      0.26     0.841     0.000      0.000
 -7.00    0.000 -0.116     1453     120      0.00      0.29     0.923     0.000      0.000
 -8.00    0.000 -0.167     2089     171      0.00      0.30     0.886     0.000      0.000
 -9.00    0.000 -0.246     3073     253      0.01      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
            TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  0.00000E+00, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
           SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
           ENDPOINT GLOBAL ERROR= 1.000E-02 *(TOL** 1.000) APPROX,      R.M.S. RESIDUAL= 0.0E+00 OVER  9 VALUES
           MAXIMUM  GLOBAL ERROR= 1.985E-02 *(TOL** 0.830) APPROX,      R.M.S. RESIDUAL= 1.4E-01 OVER  8 VALUES
           MAXIMUM DEFECT       = 0.743     *(TOL** 0.986) APPROX,      R.M.S. RESIDUAL= 4.5E-02 OVER  8 VALUES
NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
ACCURACY LOG10 TOL                 CALLS   STEPS
10** -2   -1.90    0.000 -0.056      698      57
10** -3   -2.91    0.000 -0.058      729      59
10** -4   -3.93    0.000 -0.064      799      64
10** -5   -4.94    0.000 -0.073      917      72
10** -6   -5.96    0.000 -0.090     1120      90
10** -7   -6.97    0.000 -0.115     1443     119
```

```
10** -8   -7.99    0.000 -0.166     2079     170


C3   (SCALED)

 LOG10     TIME   OVHD     FCN    NO OF   END PNT   MAXIMUM   MAXIMUM   FRACTION   FRACTION
  TOL                     CALLS   STEPS   GLB ERR   GLB ERR   DEFECT    DECEIVED   BAD DECV
 -1.00    0.000 -0.030      325      26      0.01      0.06     0.856     0.000      0.000
 -2.00    0.000 -0.032      349      28      0.00      0.06     0.919     0.000      0.000
 -3.00    0.000 -0.037      397      31      0.00      0.06     0.883     0.000      0.000
 -4.00    0.000 -0.042      457      36      0.00      0.10     0.943     0.000      0.000
 -5.00    0.000 -0.052      565      44      0.01      0.31     0.772     0.000      0.000
 -6.00    0.000 -0.064      697      57      0.28      0.35     0.624     0.000      0.000
 -7.00    0.000 -0.093     1009      84      0.49      0.61     0.625     0.000      0.000
 -8.00    0.000 -0.140     1525     127      0.63      0.74     0.596     0.000      0.000
 -9.00    0.000 -0.215     2341     195      0.74      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
              TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  1.80313E-04, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
           SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
           ENDPOINT GLOBAL ERROR= 1.929E-03 *(TOL** 0.705) APPROX,      R.M.S. RESIDUAL= 3.8E-01 OVER  9 VALUES
           MAXIMUM  GLOBAL ERROR= 2.513E-02 *(TOL** 0.812) APPROX,      R.M.S. RESIDUAL= 1.3E-01 OVER  8 VALUES
           MAXIMUM DEFECT       =  1.04     *(TOL** 1.030) APPROX,      R.M.S. RESIDUAL= 3.8E-02 OVER  8 VALUES
    NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
    EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
    ACCURACY LOG10 TOL                 CALLS   STEPS
    10** -1   -0.99    0.000 -0.030      324      25
```

10** -2	-1.96	0.000	-0.032	348	27
10** -3	-2.93	0.000	-0.036	393	30
10** -4	-3.90	0.000	-0.041	451	35
10** -5	-4.87	0.000	-0.051	551	42
10** -6	-5.84	0.000	-0.062	676	54
10** -7	-6.82	0.000	-0.087	951	79
10** -8	-7.79	0.000	-0.130	1415	117

C4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.077	361	28	0.01	0.06	0.854	0.000	0.000	
-2.00	0.000	-0.082	385	30	0.04	0.06	0.925	0.000	0.000	
-3.00	0.000	-0.093	433	33	0.38	0.41	0.930	0.000	0.000	
-4.00	0.000	-0.105	493	37	4.27	4.49	0.814	0.000	0.000	
-5.00	0.000	-0.128	601	45	50.60	53.26	0.825	0.000	0.000	
-6.00	0.000	-0.177	829	64	307.75	308.29	0.969	0.000	0.000	
-7.00	0.000	-0.259	1213	98	603.76	604.21	0.967	0.000	0.000	
-8.00	0.000	-0.390	1825	152	805.87	806.74	0.822	0.000	0.000	
-9.00	0.000	-0.611	2857	238	930.87	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 1.80313E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 4.495E-03 *(TOL** 0.316) APPROX, R.M.S. RESIDUAL= 4.9E-01 OVER 9 VALUES

```
MAXIMUM  GLOBAL ERROR= 6.454E-03 *(TOL** 0.300) APPROX,      R.M.S. RESIDUAL= 3.6E-01 OVER  8 VALUES

MAXIMUM DEFECT       = 0.882     *(TOL** 1.000) APPROX,      R.M.S. RESIDUAL= 3.0E-02 OVER  8 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.95	0.000	-0.077	359	27
10** -2	-1.95	0.000	-0.082	383	29
10** -3	-2.95	0.000	-0.092	430	32
10** -4	-3.95	0.000	-0.105	489	36
10** -5	-4.95	0.000	-0.127	595	44
10** -6	-5.95	0.000	-0.175	817	63
10** -7	-6.95	0.000	-0.255	1193	96
10** -8	-7.95	0.000	-0.383	1793	149

C5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.077	49	4	0.39	0.42	0.048	0.000	0.000
-2.00	0.000	-0.077	49	4	3.88	3.89	0.479	0.000	0.000
-3.00	0.000	-0.096	61	5	4.97	4.98	0.375	0.000	0.000
-4.00	0.000	-0.153	97	7	6.74	6.77	0.630	0.000	0.000
-5.00	0.000	-0.171	109	9	8.55	8.56	0.692	0.000	0.000
-6.00	0.000	-0.247	157	13	8.65	8.66	0.658	0.000	0.000

-7.00	0.000	-0.379	241	20	7.31	7.35	0.640	0.000	0.000
-8.00	0.000	-0.549	349	29	6.94	7.01	0.641	0.000	0.000
-9.00	0.000	-0.851	541	45	6.50	6.54	0.683	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.47   *(TOL** 0.898) APPROX,   R.M.S. RESIDUAL= 3.0E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 1.52   *(TOL** 0.900) APPROX,   R.M.S. RESIDUAL= 2.9E-01 OVER 9 VALUES

MAXIMUM DEFECT   = 0.156   *(TOL** 0.909) APPROX,   R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY -   MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.31	0.000	-0.077	49	4
10** -3	-2.41	0.000	-0.085	53	4
10** -4	-3.51	0.000	-0.125	79	6
10** -5	-4.61	0.000	-0.164	104	8
10** -6	-5.71	0.000	-0.225	143	11
10** -7	-6.81	0.000	-0.354	225	18
10** -8	-7.92	0.000	-0.534	339	28
10** -9	-9.02	0.000	-0.855	544	45

SUMMARY OVER GROUP 3

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.248	1613	128	0.39	0.42	0.856	0.000	0.000

-2.00	0.000	-0.256	1673	135	3.88	3.89	0.925	0.000	0.000
-3.00	0.000	-0.293	1841	147	4.97	4.98	0.930	0.000	0.000
-4.00	0.000	-0.378	2153	170	6.74	6.77	0.982	0.000	0.000
-5.00	0.000	-0.446	2645	208	50.60	53.26	0.994	0.000	0.000
-6.00	0.000	-0.608	3461	279	307.75	308.29	0.969	0.000	0.000
-7.00	0.000	-0.892	4901	404	603.76	604.21	0.967	0.000	0.000
-8.00	0.000	-1.314	7289	604	805.87	806.74	0.886	0.000	0.000
-9.00	0.000	-2.028	11141	925	930.87	6.54	0.683	0.000	0.000

(LOC ASSESS ON 239)

GROUP 4 CRK45 INT=a3

D1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.037	421	28	16.10	20.08	0.973	0.000	0.000
-2.00	0.000	-0.034	385	28	74.00	82.12	0.971	0.000	0.000
-3.00	0.000	-0.048	541	39	57.62	67.08	0.884	0.000	0.000
-4.00	0.000	-0.068	769	57	40.42	46.96	0.914	0.000	0.000
-5.00	0.000	-0.088	997	83	28.93	33.34	0.954	0.000	0.000
-6.00	0.000	-0.131	1489	124	11.61	13.30	0.834	0.000	0.000
-7.00	0.000	-0.197	2245	187	3.44	4.00	0.814	0.000	0.000
-8.00	0.000	-0.304	3457	288	15.22	17.52	0.671	0.000	0.000

-9.00	0.000	-0.474	5389	449	22.37	25.84	0.644	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 51.4 *(TOL** 1.075) APPROX, R.M.S. RESIDUAL= 3.3E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 60.7 *(TOL** 1.076) APPROX, R.M.S. RESIDUAL= 3.2E-01 OVER 9 VALUES

MAXIMUM DEFECT = 1.08 *(TOL** 1.022) APPROX, R.M.S. RESIDUAL= 2.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-1.01	0.000	-0.037	420	28
10** -2	-1.99	0.000	-0.034	385	28
10** -3	-2.97	0.000	-0.047	536	38
10** -4	-3.95	0.000	-0.067	757	56
10** -5	-4.93	0.000	-0.086	980	81
10** -6	-5.91	0.000	-0.127	1442	120
10** -7	-6.88	0.000	-0.190	2157	179
10** -8	-7.86	0.000	-0.289	3290	274
10** -9	-8.84	0.000	-0.447	5082	423

D2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.039	445	27	20.64	20.64	0.880	0.000	0.000
-2.00	0.000	-0.045	517	32	28.78	63.81	0.835	0.000	0.000

-3.00	0.000	-0.063	721	44	26.29	58.35	0.842	0.000	0.000
-4.00	0.000	-0.088	997	63	35.59	75.20	0.912	0.000	0.000
-5.00	0.000	-0.123	1393	91	31.97	65.20	0.979	0.000	0.000
-6.00	0.000	-0.171	1945	132	36.86	75.07	1.000	0.008	0.000
-7.00	0.000	-0.205	2329	194	30.53	62.61	0.955	0.000	0.000
-8.00	0.000	-0.317	3601	300	3.18	6.36	0.773	0.000	0.000
-9.00	0.000	-0.495	5629	469	11.17	21.64	0.701	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 42.9 *(TOL** 1.063) APPROX, R.M.S. RESIDUAL= 2.8E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 68.7 *(TOL** 1.048) APPROX, R.M.S. RESIDUAL= 3.3E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.930 *(TOL** 1.006) APPROX, R.M.S. RESIDUAL= 4.5E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.96	0.000	-0.039	442	26
10** -2	-1.96	0.000	-0.045	513	31
10** -3	-2.95	0.000	-0.063	711	43
10** -4	-3.95	0.000	-0.086	982	61
10** -5	-4.94	0.000	-0.120	1369	89
10** -6	-5.93	0.000	-0.168	1908	129
10** -7	-6.93	0.000	-0.202	2301	189
10** -8	-7.92	0.000	-0.308	3502	291
10** -9	-8.92	0.000	-0.480	5460	454

D3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.037	421	26	5.26	21.46	0.906	0.000	0.000
-2.00	0.000	-0.059	673	39	4.84	35.00	0.953	0.000	0.000
-3.00	0.000	-0.074	841	53	6.08	41.81	0.994	0.000	0.000
-4.00	0.000	-0.105	1189	75	14.51	80.52	0.816	0.000	0.000
-5.00	0.000	-0.146	1657	107	20.24	101.38	0.989	0.000	0.000
-6.00	0.000	-0.209	2377	154	20.86	105.53	0.963	0.000	0.000
-7.00	0.000	-0.279	3169	223	28.67	143.93	0.975	0.000	0.000
-8.00	0.000	-0.349	3973	331	28.41	145.10	0.912	0.000	0.000
-9.00	0.000	-0.544	6181	515	7.18	0.00	0.000	*********	********** (LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 2.03218E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 5.25 *(TOL** 0.927) APPROX, R.M.S. RESIDUAL= 2.4E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 19.9 *(TOL** 0.879) APPROX, R.M.S. RESIDUAL= 7.6E-02 OVER 8 VALUES

MAXIMUM DEFECT = 0.924 *(TOL** 0.999) APPROX, R.M.S. RESIDUAL= 2.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.036	412	25
10** -2	-1.97	0.000	-0.058	665	38

10** -3	-2.97	0.000	-0.074	835	52
10** -4	-3.97	0.000	-0.104	1178	74
10** -5	-4.97	0.000	-0.145	1644	106
10** -6	-5.97	0.000	-0.207	2357	152
10** -7	-6.97	0.000	-0.277	3149	221
10** -8	-7.98	0.000	-0.348	3953	328

D4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.063	721	38	4.82	19.78	0.669	0.000	0.000	
-2.00	0.000	-0.081	925	51	3.41	70.07	0.996	0.000	0.000	
-3.00	0.000	-0.095	1081	67	3.30	49.36	1.382	0.060	0.000	
-4.00	0.000	-0.133	1513	95	6.71	132.70	0.944	0.000	0.000	
-5.00	0.000	-0.185	2101	134	12.05	210.56	0.990	0.000	0.000	
-6.00	0.000	-0.263	2989	192	13.71	201.68	1.002	0.005	0.000	
-7.00	0.000	-0.365	4153	277	16.71	255.02	1.003	0.007	0.000	
-8.00	0.000	-0.428	4861	402	32.11	511.92	1.008	0.007	0.000	
-9.00	0.000	-0.655	7441	620	16.90	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 6.35521E-05, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 2.53 *(TOL** 0.886) APPROX, R.M.S. RESIDUAL= 1.4E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 19.9 *(TOL** 0.825) APPROX, R.M.S. RESIDUAL= 1.3E-01 OVER 8 VALUES

MAXIMUM DEFECT = 0.883 *(TOL** 0.990) APPROX, R.M.S. RESIDUAL= 7.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.96	0.000	-0.063	712	37
10** -2	-1.97	0.000	-0.081	918	50
10** -3	-2.98	0.000	-0.095	1077	66
10** -4	-3.99	0.000	-0.133	1507	94
10** -5	-5.00	0.000	-0.185	2099	133
10** -6	-6.01	0.000	-0.264	2998	192
10** -7	-7.02	0.000	-0.366	4166	279
10** -8	-8.03	0.000	-0.429	4881	405

D5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.077	877	47	17.97	19.61	0.933	0.000	0.000	(LOC ASSESS ON 44)
TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 1.89488E+01, ERROR FLAG (GLOBAL) -1, (LOCAL) 2										
-2.00	0.000	-0.130	1477	81	0.27	44.26	1.018	0.012	0.000	
-3.00	0.000	-0.162	1837	106	2.57	211.84	0.765	0.000	0.000	
-4.00	0.000	-0.205	2329	144	3.96	391.30	1.604	0.021	0.000	
-5.00	0.000	-0.279	3169	201	8.89	191.54	2.858	0.020	0.000	
-6.00	0.000	-0.389	4417	285	24.70	1718.05	1.181	0.014	0.000	

```
  -7.00    0.000 -0.549     6241     414     31.87   2649.58     1.029     0.005      0.000
  -8.00    0.000 -0.677     7693     613     42.07      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
              TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  6.78306E-05, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
  -9.00    0.000 -0.999    11353     946     29.13      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
              TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  0.00000E+00, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
           SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
           ENDPOINT GLOBAL ERROR=  1.23     *(TOL** 0.826) APPROX,      R.M.S. RESIDUAL= 5.0E-01 OVER  9 VALUES
           MAXIMUM  GLOBAL ERROR=  13.5     *(TOL** 0.677) APPROX,      R.M.S. RESIDUAL= 2.4E-01 OVER  6 VALUES
           MAXIMUM DEFECT      = 0.996     *(TOL** 0.976) APPROX,      R.M.S. RESIDUAL= 1.8E-01 OVER  6 VALUES
 NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
 EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
 ACCURACY LOG10 TOL                 CALLS   STEPS
 10** -2   -2.05    0.000 -0.131     1494      82
 10** -3   -3.07    0.000 -0.165     1872     108
 10** -4   -4.10    0.000 -0.212     2410     149
 10** -5   -5.12    0.000 -0.292     3320     211
 10** -6   -6.15    0.000 -0.412     4682     303


SUMMARY OVER GROUP  4
   LOG10     TIME   OVHD      FCN     NO OF   END PNT   MAXIMUM   MAXIMUM   FRACTION   FRACTION
    TOL                      CALLS    STEPS   GLB ERR   GLB ERR   DEFECT    DECEIVED   BAD DECV
  -1.00    0.000 -0.254     2885     166     20.64     21.46     0.973     0.000      0.000
                                                                                       (LOC ASSESS ON 163)
```

-2.00	0.000	-0.350	3977	231	74.00	82.12	1.018	0.004	0.000	
-3.00	0.000	-0.442	5021	309	57.62	211.84	1.382	0.013	0.000	
-4.00	0.000	-0.598	6797	434	40.42	391.30	1.604	0.007	0.000	
-5.00	0.000	-0.820	9317	616	31.97	210.56	2.858	0.006	0.000	
-6.00	0.000	-1.163	13217	887	36.86	1718.05	1.181	0.007	0.000	
-7.00	0.000	-1.595	18137	1295	31.87	2649.58	1.029	0.003	0.000	
-8.00	0.000	-2.075	23585	1934	42.07	511.92	1.008	0.002	0.000	
										(LOC ASSESS ON1321)
-9.00	0.000	-3.166	35993	2999	29.13	25.84	0.701	0.000	0.000	
										(LOC ASSESS ON 918)

GROUP 5 CRK45 INT=a3

E1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.006	121	9	3.02	3.02	0.899	0.000	0.000
-2.00	0.000	-0.008	157	13	3.07	3.07	0.874	0.000	0.000
-3.00	0.000	-0.012	241	20	2.06	3.14	0.836	0.000	0.000
-4.00	0.000	-0.019	361	30	2.39	3.34	0.883	0.000	0.000
-5.00	0.000	-0.028	553	46	2.56	3.54	0.863	0.000	0.000
-6.00	0.000	-0.044	865	72	2.47	3.50	0.784	0.000	0.000
-7.00	0.000	-0.069	1345	112	2.29	3.51	0.723	0.000	0.000

-8.00	0.000	-0.109	2125	177	2.58	3.51	0.680	0.000	0.000
-9.00	0.000	-0.172	3349	279	2.83	3.51	0.648	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 2.68   *(TOL** 1.004) APPROX,   R.M.S. RESIDUAL= 5.2E-02 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 3.01   *(TOL** 0.991) APPROX,   R.M.S. RESIDUAL= 1.2E-02 OVER 9 VALUES

MAXIMUM DEFECT   = 0.975   *(TOL** 1.018) APPROX,   R.M.S. RESIDUAL= 1.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY -   MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.006	119	8
10** -2	-1.95	0.000	-0.008	155	12
10** -3	-2.94	0.000	-0.012	235	19
10** -4	-3.92	0.000	-0.018	351	29
10** -5	-4.90	0.000	-0.027	534	44
10** -6	-5.88	0.000	-0.042	828	68
10** -7	-6.87	0.000	-0.066	1280	106
10** -8	-7.85	0.000	-0.103	2007	167
10** -9	-8.83	0.000	-0.161	3142	261

E2   (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.026	721	39	0.23	0.54	0.989	0.000	0.000

-2.00	0.000	-0.034	937	49	0.33	0.84	0.740	0.000	0.000
-3.00	0.000	-0.044	1225	66	0.47	1.02	0.500	0.000	0.000
-4.00	0.000	-0.057	1585	90	1.30	3.15	0.968	0.000	0.000
-5.00	0.000	-0.070	1957	122	1.66	4.11	0.965	0.000	0.000
-6.00	0.000	-0.101	2797	174	1.85	4.67	1.342	0.023	0.000
-7.00	0.000	-0.138	3829	250	2.03	5.12	1.272	0.012	0.000
-8.00	0.000	-0.179	4981	365	1.82	4.61	1.241	0.014	0.000
-9.00	0.000	-0.246	6853	557	1.29	3.35	1.019	0.002	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 0.265 *(TOL** 0.889) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.621 *(TOL** 0.884) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.701 *(TOL** 0.972) APPROX, R.M.S. RESIDUAL= 1.0E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.90	0.000	-0.033	915	47
10** -3	-2.93	0.000	-0.043	1204	64
10** -4	-3.96	0.000	-0.056	1569	88
10** -5	-4.98	0.000	-0.070	1951	121
10** -6	-6.01	0.000	-0.101	2811	175
10** -7	-7.04	0.000	-0.139	3877	254
10** -8	-8.07	0.000	-0.184	5114	378
10** -9	-9.10	0.000	-0.253	7040	576

E3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.038	409	23	2.78	2.83	0.812	0.000	0.000
-2.00	0.000	-0.062	673	35	0.41	0.53	0.823	0.000	0.000
-3.00	0.000	-0.079	865	48	3.18	3.23	0.846	0.000	0.000
-4.00	0.000	-0.098	1069	66	3.69	3.77	0.950	0.000	0.000
-5.00	0.000	-0.132	1441	94	2.32	2.50	0.999	0.000	0.000
-6.00	0.000	-0.182	1981	136	1.35	1.73	0.958	0.000	0.000
-7.00	0.000	-0.246	2677	202	1.44	1.87	0.985	0.000	0.000
-8.00	0.000	-0.343	3733	302	1.73	2.22	0.995	0.000	0.000
-9.00	0.000	-0.509	5545	459	1.81	1.97	0.962	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.80 *(TOL** 1.000) APPROX, R.M.S. RESIDUAL= 2.7E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 1.89 *(TOL** 0.993) APPROX, R.M.S. RESIDUAL= 2.3E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.810 *(TOL** 0.989) APPROX, R.M.S. RESIDUAL= 1.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.92	0.000	-0.036	387	22
10** -2	-1.93	0.000	-0.060	654	34
10** -3	-2.94	0.000	-0.078	853	47

10** -4	-3.95	0.000	-0.097	1059	65
10** -5	-4.96	0.000	-0.131	1427	93
10** -6	-5.98	0.000	-0.181	1968	134
10** -7	-6.99	0.000	-0.245	2668	201
10** -8	-8.00	0.000	-0.343	3731	301
10** -9	-9.01	0.000	-0.511	5563	460

E4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.00	0.01	0.004	0.000	0.000
-2.00	0.000	-0.002	49	4	0.01	0.05	0.044	0.000	0.000
-3.00	0.000	-0.002	49	4	0.11	0.54	0.439	0.000	0.000
-4.00	0.000	-0.003	73	5	0.08	0.39	0.351	0.000	0.000
-5.00	0.000	-0.003	85	6	0.18	0.69	0.584	0.000	0.000
-6.00	0.000	-0.005	133	8	0.29	0.57	0.630	0.000	0.000
-7.00	0.000	-0.008	193	12	0.58	0.85	0.682	0.000	0.000
-8.00	0.000	-0.011	265	17	0.80	0.88	0.710	0.000	0.000
-9.00	0.000	-0.014	349	27	0.84	1.23	0.985	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 7.438E-03 *(TOL** 0.746) APPROX,   R.M.S. RESIDUAL= 2.1E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 2.931E-02 *(TOL** 0.791) APPROX,   R.M.S. RESIDUAL= 3.6E-01 OVER 9 VALUES

MAXIMUM DEFECT   = 2.711E-02 *(TOL** 0.796) APPROX,   R.M.S. RESIDUAL= 3.5E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.80	0.000	-0.002	49	4
10** -4	-3.06	0.000	-0.002	50	4
10** -5	-4.31	0.000	-0.003	76	5
10** -6	-5.57	0.000	-0.004	112	7
10** -7	-6.82	0.000	-0.007	182	11
10** -8	-8.08	0.000	-0.011	271	17

E5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.003	49	4	0.00	0.00	0.005	0.000	0.000
-2.00	0.000	-0.003	49	4	0.01	0.01	0.052	0.000	0.000
-3.00	0.000	-0.003	49	4	0.07	0.07	0.517	0.000	0.000
-4.00	0.000	-0.004	73	5	0.08	0.08	0.372	0.000	0.000
-5.00	0.000	-0.006	97	6	0.50	0.50	0.423	0.000	0.000
-6.00	0.000	-0.007	121	7	0.93	0.93	0.678	0.000	0.000
-7.00	0.000	-0.012	193	10	1.52	1.53	0.556	0.000	0.000
-8.00	0.000	-0.018	289	14	2.25	2.28	1.002	0.071	0.000
-9.00	0.000	-0.021	349	20	2.82	2.89	1.019	0.150	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

```
ENDPOINT GLOBAL ERROR= 4.503E-03 *(TOL** 0.656) APPROX,     R.M.S. RESIDUAL= 2.2E-01 OVER  9 VALUES
MAXIMUM  GLOBAL ERROR= 4.542E-03 *(TOL** 0.655) APPROX,     R.M.S. RESIDUAL= 2.2E-01 OVER  9 VALUES
MAXIMUM DEFECT       = 2.815E-02 *(TOL** 0.796) APPROX,     R.M.S. RESIDUAL= 3.5E-01 OVER  9 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.82	0.000	-0.003	49	4
10** -4	-3.08	0.000	-0.003	50	4
10** -5	-4.33	0.000	-0.005	80	5
10** -6	-5.59	0.000	-0.007	111	6
10** -7	-6.84	0.000	-0.011	181	9
10** -8	-8.10	0.000	-0.018	294	14

SUMMARY OVER GROUP 5

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.075	1349	79	3.02	3.02	0.989	0.000	0.000
-2.00	0.000	-0.108	1865	105	3.07	3.07	0.874	0.000	0.000
-3.00	0.000	-0.141	2429	142	3.18	3.23	0.846	0.000	0.000
-4.00	0.000	-0.181	3161	196	3.69	3.77	0.968	0.000	0.000
-5.00	0.000	-0.240	4133	274	2.56	4.11	0.999	0.000	0.000
-6.00	0.000	-0.339	5897	397	2.47	4.67	1.342	0.010	0.000
-7.00	0.000	-0.472	8237	586	2.29	5.12	1.272	0.005	0.000

-8.00	0.000	-0.659	11393	875	2.58	4.61	1.241	0.007	0.000
-9.00	0.000	-0.962	16445	1342	2.83	3.51	1.019	0.003	0.000

SUMMARY OVER ALL GROUPS CRK45 INT=a3

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.663	7368	470	0.989	0.000	0.000	(LOC ASSESS ON 467)
-2.00	0.000	-0.839	9853	623	1.018	0.003	0.000	
-3.00	0.000	-1.041	12313	796	1.382	0.006	0.000	
-4.00	0.000	-1.373	16081	1065	1.604	0.005	0.000	
-5.00	0.000	-1.803	21541	1470	2.858	0.003	0.000	
-6.00	0.000	-2.521	30037	2095	1.436	0.007	0.000	
-7.00	0.000	-3.534	41653	3060	1.272	0.004	0.000	
-8.00	0.000	-4.867	56953	4562	1.241	0.003	0.000	(LOC ASSESS ON3949)
-9.00	0.000	-7.366	85309	7023	1.694	0.005	0.000	(LOC ASSESS ON3640)
OVERALL SUMMARY	0.000	-24.007	281108	21164	2.858	0.004	0.000	

C.2 Reliability of the Estimate

NONSTIFF DETEST PACKAGE   OPTION= 5, NORMEF= 3, NRMTYP= 1, GLBDEF= 0   ON   Sun 4/280-32

GROUP 1   CRK45   INT=b3

A1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.004	109	8	0.01	0.24	1.015	0.500	0.000
-2.00	0.000	-0.004	109	9	0.01	0.18	1.016	0.556	0.000
-3.00	0.000	-0.005	133	11	0.01	0.12	1.008	0.364	0.000
-4.00	0.000	-0.007	169	14	0.12	0.12	1.014	0.214	0.000
-5.00	0.000	-0.009	241	20	0.00	0.14	1.001	0.050	0.000
-6.00	0.000	-0.013	337	28	0.00	0.16	0.998	0.000	0.000
-7.00	0.000	-0.019	493	41	0.01	0.17	0.997	0.000	0.000
-8.00	0.000	-0.028	721	60	0.01	0.19	0.993	0.000	0.000
-9.00	0.000	-0.043	1093	91	0.02	0.20	0.993	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.537E-02 *(TOL** 1.002) APPROX,   R.M.S. RESIDUAL= 3.3E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.158 *(TOL** 0.996) APPROX,   R.M.S. RESIDUAL= 9.1E-02 OVER 9 VALUES

MAXIMUM DEFECT = 0.809 *(TOL** 1.018) APPROX,   R.M.S. RESIDUAL= 4.7E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS

10** -2	-1.87	0.000	-0.004	109	8
10** -3	-2.86	0.000	-0.005	129	10
10** -4	-3.84	0.000	-0.006	163	13
10** -5	-4.82	0.000	-0.009	227	18
10** -6	-5.80	0.000	-0.012	317	26
10** -7	-6.78	0.000	-0.018	459	38
10** -8	-7.76	0.000	-0.026	667	55
10** -9	-8.75	0.000	-0.039	998	83

A2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.003	73	6	0.01	0.14	0.992	0.000	0.000
-2.00	0.000	-0.003	85	7	0.02	0.11	0.991	0.000	0.000
-3.00	0.000	-0.004	97	8	0.12	0.12	0.990	0.000	0.000
-4.00	0.000	-0.005	133	11	0.09	0.16	0.990	0.000	0.000
-5.00	0.000	-0.007	181	14	0.21	0.27	0.991	0.000	0.000
-6.00	0.000	-0.008	229	19	0.29	0.42	0.991	0.000	0.000
-7.00	0.000	-0.011	313	26	0.46	0.61	0.992	0.000	0.000
-8.00	0.000	-0.016	433	36	0.66	0.81	0.992	0.000	0.000
-9.00	0.000	-0.022	613	51	0.91	1.00	0.992	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.014E-02 *(TOL** 0.766) APPROX,   R.M.S. RESIDUAL= 1.7E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 6.568E-02 *(TOL** 0.870) APPROX, R.M.S. RESIDUAL= 9.9E-02 OVER 9 VALUES

MAXIMUM DEFECT = 0.574 *(TOL** 0.986) APPROX, R.M.S. RESIDUAL= 6.8E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.78	0.000	-0.003	82	6
10** -3	-2.80	0.000	-0.003	94	7
10** -4	-3.81	0.000	-0.005	126	10
10** -5	-4.83	0.000	-0.006	172	13
10** -6	-5.84	0.000	-0.008	221	18
10** -7	-6.85	0.000	-0.011	300	24
10** -8	-7.87	0.000	-0.015	417	34
10** -9	-8.88	0.000	-0.022	591	49

A3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.009	181	10	2.25	2.24	5.726	0.500	0.100
-2.00	0.000	-0.010	205	11	2.79	2.83	8.123	0.727	0.182
-3.00	0.000	-0.014	289	18	4.83	4.82	1.464	0.611	0.000
-4.00	0.000	-0.020	409	26	8.20	8.12	3.425	0.769	0.000
-5.00	0.000	-0.029	577	39	4.67	4.59	1.316	0.718	0.000
-6.00	0.000	-0.044	889	59	5.20	5.17	11.487	0.644	0.017

-7.00	0.000	-0.067	1357	88	5.58	5.57	22.157	0.477	0.011
-8.00	0.000	-0.095	1921	131	4.16	4.16	32.804	0.389	0.023
-9.00	0.000	-0.137	2773	197	3.10	3.09	8.942	0.355	0.015

```
SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
ENDPOINT GLOBAL ERROR=  3.49    *(TOL** 0.983) APPROX,     R.M.S. RESIDUAL= 1.6E-01 OVER  9 VALUES
MAXIMUM  GLOBAL ERROR=  3.49    *(TOL** 0.984) APPROX,     R.M.S. RESIDUAL= 1.5E-01 OVER  9 VALUES
MAXIMUM DEFECT       = 0.813    *(TOL** 0.978) APPROX,     R.M.S. RESIDUAL= 5.9E-02 OVER  9 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.93	0.000	-0.009	179	9
10** -2	-1.95	0.000	-0.010	203	10
10** -3	-2.98	0.000	-0.014	287	17
10** -4	-4.00	0.000	-0.020	408	25
10** -5	-5.02	0.000	-0.029	583	39
10** -6	-6.05	0.000	-0.045	910	60
10** -7	-7.07	0.000	-0.069	1395	90
10** -8	-8.09	0.000	-0.099	1998	137

A4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.01	0.01	1.053	0.250	0.000

-2.00	0.000	-0.002	49	4	0.06	0.12	1.053	0.250	0.000
-3.00	0.000	-0.002	49	4	0.57	1.23	1.053	0.250	0.000
-4.00	0.000	-0.004	73	5	0.73	1.48	1.056	0.200	0.000
-5.00	0.000	-0.004	85	6	2.19	4.17	1.049	0.333	0.000
-6.00	0.000	-0.005	97	8	2.82	4.92	1.392	0.250	0.000
-7.00	0.000	-0.007	145	12	2.86	5.58	1.285	0.167	0.000
-8.00	0.000	-0.011	217	17	1.93	5.38	3.233	0.353	0.000
-9.00	0.000	-0.018	361	27	1.23	5.34	1.411	0.222	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.427E-02 *(TOL** 0.751) APPROX, R.M.S. RESIDUAL= 4.8E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 4.628E-02 *(TOL** 0.712) APPROX, R.M.S. RESIDUAL= 4.6E-01 OVER 9 VALUES

MAXIMUM DEFECT = 2.797E-02 *(TOL** 0.788) APPROX, R.M.S. RESIDUAL= 3.9E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.83	0.000	-0.002	49	4
10** -4	-3.10	0.000	-0.003	51	4
10** -5	-4.37	0.000	-0.004	77	5
10** -6	-5.64	0.000	-0.005	92	7
10** -7	-6.91	0.000	-0.007	140	11
10** -8	-8.18	0.000	-0.012	242	18

A5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.01	0.01	1.367	0.750	0.000
-2.00	0.000	-0.002	49	4	0.07	0.07	1.367	0.750	0.000
-3.00	0.000	-0.002	49	4	0.72	0.72	1.367	0.750	0.000
-4.00	0.000	-0.002	61	5	2.66	2.67	1.373	0.800	0.000
-5.00	0.000	-0.003	85	7	4.08	4.22	1.422	0.714	0.000
-6.00	0.000	-0.005	121	10	5.82	5.94	1.405	0.600	0.000
-7.00	0.000	-0.007	169	14	7.25	7.54	1.311	0.643	0.000
-8.00	0.000	-0.010	241	20	8.97	9.25	1.207	0.600	0.000
-9.00	0.000	-0.015	361	30	9.43	10.12	1.113	0.600	0.000

```
SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
ENDPOINT GLOBAL ERROR= 2.645E-02 *(TOL** 0.656) APPROX,      R.M.S. RESIDUAL= 4.6E-01 OVER  9 VALUES
MAXIMUM  GLOBAL ERROR= 2.661E-02 *(TOL** 0.654) APPROX,      R.M.S. RESIDUAL= 4.5E-01 OVER  9 VALUES
MAXIMUM DEFECT       = 9.608E-03 *(TOL** 0.734) APPROX,      R.M.S. RESIDUAL= 3.9E-01 OVER  9 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.34	0.000	-0.002	49	4
10** -4	-2.70	0.000	-0.002	49	4
10** -5	-4.07	0.000	-0.003	62	5
10** -6	-5.43	0.000	-0.004	100	8
10** -7	-6.79	0.000	-0.006	159	13

10** -8	-8.16	0.000	-0.010	259	21

SUMMARY OVER GROUP 1

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.020	461	32	2.25	2.24	5.726	0.406	0.031
-2.00	0.000	-0.022	497	35	2.79	2.83	8.123	0.486	0.057
-3.00	0.000	-0.027	617	45	4.83	4.82	1.464	0.422	0.000
-4.00	0.000	-0.038	845	61	8.20	8.12	3.425	0.459	0.000
-5.00	0.000	-0.052	1169	86	4.67	4.59	1.422	0.419	0.000
-6.00	0.000	-0.075	1673	124	5.82	5.94	11.487	0.371	0.008
-7.00	0.000	-0.112	2477	181	7.25	7.54	22.157	0.293	0.006
-8.00	0.000	-0.159	3533	264	8.97	9.25	32.804	0.261	0.011
-9.00	0.000	-0.235	5201	396	9.43	10.12	8.942	0.237	0.008

GROUP 2 CRK45 INT=b3

B1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	**************		2257	100	********	113370.52	229.259	0.850	0.140
	METHOD FAILED AT X = 1.63639E+01								
-2.00	0.000	-0.031	661	38	4.83	16.93	3.104	0.184	0.000

```
 -3.00    0.000 -0.040      853       50       9.78     40.81     1.474     0.240      0.000
 -4.00    0.000 -0.056     1213       70       5.88     24.08     4.703     0.257      0.000
 -5.00    0.000 -0.075     1609       99       8.67     38.86     1.119     0.111      0.000
 -6.00    0.000 -0.101     2161      144      11.67     51.60     1.076     0.125      0.000
 -7.00    0.000 -0.132     2845      211      11.74     51.25     1.084     0.090      0.000
 -8.00    0.000 -0.185     3973      314      13.93     59.57     1.007     0.029      0.000
 -9.00    0.000 -0.272     5845      478      15.44      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
             TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  3.90652E-03, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
          SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
          ENDPOINT GLOBAL ERROR=  4.29     *(TOL** 0.937) APPROX,      R.M.S. RESIDUAL= 8.0E-02 OVER  8 VALUES
          MAXIMUM  GLOBAL ERROR=  15.3     *(TOL** 0.923) APPROX,      R.M.S. RESIDUAL= 9.9E-02 OVER  7 VALUES
          MAXIMUM DEFECT       = 0.811     *(TOL** 0.989) APPROX,      R.M.S. RESIDUAL= 2.7E-02 OVER  7 VALUES
NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
ACCURACY LOG10 TOL                 CALLS   STEPS
10** -2   -1.93    0.000 -0.030      647      37
10** -3   -2.94    0.000 -0.039      841      49
10** -4   -3.95    0.000 -0.056     1195      69
10** -5   -4.96    0.000 -0.074     1594      97
10** -6   -5.97    0.000 -0.100     2147     142
10** -7   -6.99    0.000 -0.132     2835     210
10** -8   -8.00    0.000 -0.185     3969     313
```

B2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.009	253	20	0.01	0.07	1.016	0.850	0.000
-2.00	0.000	-0.010	277	22	0.00	0.07	1.014	0.727	0.000
-3.00	0.000	-0.012	325	25	0.00	0.08	1.014	0.560	0.000
-4.00	0.000	-0.013	361	29	0.00	0.08	1.013	0.448	0.000
-5.00	0.000	-0.016	445	35	0.01	0.11	1.013	0.343	0.000
-6.00	0.000	-0.021	577	45	0.01	0.13	1.029	0.156	0.000
-7.00	0.000	-0.028	757	62	0.00	0.15	1.041	0.048	0.000
-8.00	0.000	-0.040	1093	91	0.01	0.17	0.994	0.000	0.000
-9.00	0.000	-0.061	1657	138	0.02	0.00	0.000	*********	********** (LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 2.58297E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.053E-02 *(TOL** 0.994) APPROX, R.M.S. RESIDUAL= 7.3E-02 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 5.549E-02 *(TOL** 0.942) APPROX, R.M.S. RESIDUAL= 3.1E-02 OVER 8 VALUES

MAXIMUM DEFECT = 1.07 *(TOL** 1.028) APPROX, R.M.S. RESIDUAL= 5.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-1.00	0.000	-0.009	253	20
10** -2	-1.97	0.000	-0.010	276	21
10** -3	-2.95	0.000	-0.012	322	24

10** -4	-3.92	0.000	-0.013	358	28
10** -5	-4.89	0.000	-0.016	435	34
10** -6	-5.86	0.000	-0.021	558	43
10** -7	-6.84	0.000	-0.027	727	59
10** -8	-7.81	0.000	-0.038	1028	85

B3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.004	121	9	0.01	0.20	1.012	0.444	0.000
-2.00	0.000	-0.005	145	11	0.02	0.09	1.012	0.364	0.000
-3.00	0.000	-0.006	193	14	0.01	0.10	1.008	0.286	0.000
-4.00	0.000	-0.008	229	18	0.01	0.12	1.010	0.167	0.000
-5.00	0.000	-0.011	337	25	0.03	0.12	1.021	0.120	0.000
-6.00	0.000	-0.015	445	34	0.09	0.13	1.016	0.059	0.000
-7.00	0.000	-0.021	625	49	0.12	0.15	1.019	0.020	0.000
-8.00	0.000	-0.029	877	72	0.34	0.34	0.998	0.000	0.000
-9.00	0.000	-0.046	1381	115	0.54	0.54	0.995	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.888E-03 *(TOL** 0.779) APPROX, R.M.S. RESIDUAL= 2.3E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 7.902E-02 *(TOL** 0.936) APPROX, R.M.S. RESIDUAL= 1.8E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.530 *(TOL** 0.978) APPROX, R.M.S. RESIDUAL= 9.3E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.76	0.000	-0.005	139	10
10** -3	-2.79	0.000	-0.006	182	13
10** -4	-3.81	0.000	-0.007	222	17
10** -5	-4.83	0.000	-0.011	318	23
10** -6	-5.85	0.000	-0.014	429	32
10** -7	-6.88	0.000	-0.020	602	47
10** -8	-7.90	0.000	-0.028	851	69
10** -9	-8.92	0.000	-0.045	1340	111

B4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.038	373	20	6.48	6.44	0.999	0.000	0.000
-2.00	0.000	-0.040	385	24	4.11	4.07	1.089	0.083	0.000
-3.00	0.000	-0.057	553	35	1.85	1.71	0.990	0.000	0.000
-4.00	0.000	-0.072	697	48	1.50	1.94	0.996	0.000	0.000
-5.00	0.000	-0.100	973	71	2.81	3.68	0.992	0.000	0.000
-6.00	0.000	-0.143	1393	105	1.06	2.16	0.991	0.000	0.000
-7.00	0.000	-0.203	1981	155	1.62	2.44	0.991	0.000	0.000
-8.00	0.000	-0.298	2905	232	1.99	2.46	0.990	0.000	0.000
-9.00	0.000	-0.440	4285	354	2.56	2.69	1.005	0.003	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 3.96 *(TOL** 1.047) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 3.99 *(TOL** 1.030) APPROX, R.M.S. RESIDUAL= 1.5E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.922 *(TOL** 0.998) APPROX, R.M.S. RESIDUAL= 2.8E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.038	372	19
10** -2	-1.97	0.000	-0.039	384	23
10** -3	-2.97	0.000	-0.056	547	34
10** -4	-3.97	0.000	-0.071	692	47
10** -5	-4.97	0.000	-0.099	965	70
10** -6	-5.97	0.000	-0.142	1382	104
10** -7	-6.98	0.000	-0.202	1966	153
10** -8	-7.98	0.000	-0.296	2884	230
10** -9	-8.98	0.000	-0.437	4256	351

B5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.015	313	16	1.25	1.30	0.998	0.000	0.000
-2.00	0.000	-0.017	373	22	0.53	1.19	1.050	0.227	0.000
-3.00	0.000	-0.022	481	29	0.43	0.41	1.294	0.172	0.000

-4.00	0.000	-0.029	625	39	3.09	3.98	1.341	0.128	0.000
-5.00	0.000	-0.042	913	56	0.90	1.38	1.552	0.196	0.000
-6.00	0.000	-0.056	1213	80	1.85	2.32	1.666	0.150	0.000
-7.00	0.000	-0.078	1693	117	1.87	2.48	1.216	0.154	0.000
-8.00	0.000	-0.107	2305	176	4.34	5.94	1.108	0.159	0.000
-9.00	0.000	-0.156	3361	276	8.73	11.84	1.012	0.062	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 0.422 *(TOL** 0.880) APPROX, R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.554 *(TOL** 0.879) APPROX, R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.764 *(TOL** 0.974) APPROX, R.M.S. RESIDUAL= 1.1E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.91	0.000	-0.014	307	15
10** -2	-1.93	0.000	-0.017	368	21
10** -3	-2.96	0.000	-0.022	476	28
10** -4	-3.99	0.000	-0.029	622	38
10** -5	-5.01	0.000	-0.042	916	56
10** -6	-6.04	0.000	-0.057	1231	81
10** -7	-7.06	0.000	-0.080	1732	120
10** -8	-8.09	0.000	-0.111	2400	184

SUMMARY OVER GROUP 2

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.066	1060	65	6.48	6.44	1.016	0.323	0.000	
-2.00	0.000	-0.103	1841	117	4.83	16.93	3.104	0.291	0.000	
-3.00	0.000	-0.137	2405	153	9.78	40.81	1.474	0.229	0.000	
-4.00	0.000	-0.178	3125	204	5.88	24.08	4.703	0.191	0.000	
-5.00	0.000	-0.245	4277	286	8.67	38.86	1.552	0.129	0.000	
-6.00	0.000	-0.336	5789	408	11.67	51.60	1.666	0.096	0.000	
-7.00	0.000	-0.463	7901	594	11.74	51.25	1.216	0.069	0.000	
-8.00	0.000	-0.659	11153	885	13.93	59.57	1.108	0.042	0.000	
-9.00	0.000	-0.975	16529	1361	15.44	11.84	1.012	0.024	0.000	(LOC ASSESS ON 745)

GROUP 3 CRK45 INT=b3

C1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.008	169	13	0.00	0.10	1.019	0.462	0.000
-2.00	0.000	-0.009	193	15	0.00	0.15	1.009	0.333	0.000
-3.00	0.000	-0.010	217	18	0.05	0.27	1.022	0.111	0.000
-4.00	0.000	-0.014	301	25	0.12	0.36	0.993	0.000	0.000
-5.00	0.000	-0.020	445	37	0.08	0.48	0.993	0.000	0.000
-6.00	0.000	-0.030	649	54	0.09	0.55	0.993	0.000	0.000

-7.00	0.000	-0.045	985	82	0.11	0.59	0.991	0.000	0.000
-8.00	0.000	-0.068	1501	125	0.18	0.62	0.990	0.000	0.000
-9.00	0.000	-0.106	2329	194	0.26	0.64	0.990	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 9.257E-03 *(TOL** 0.831) APPROX,   R.M.S. RESIDUAL= 2.0E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.115   *(TOL** 0.902) APPROX,   R.M.S. RESIDUAL= 1.0E-01 OVER 9 VALUES

MAXIMUM DEFECT   = 0.687   *(TOL** 1.007) APPROX,   R.M.S. RESIDUAL= 2.8E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY -   MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.82	0.000	-0.009	188	14
10** -3	-2.82	0.000	-0.010	212	17
10** -4	-3.81	0.000	-0.013	285	23
10** -5	-4.81	0.000	-0.019	416	34
10** -6	-5.80	0.000	-0.028	607	50
10** -7	-6.79	0.000	-0.042	915	76
10** -8	-7.79	0.000	-0.063	1390	115
10** -9	-8.78	0.000	-0.098	2145	178

C2   (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.057	709	57	0.00	0.02	1.016	0.965	0.000

```
 -2.00    0.000 -0.056      697      58      0.00      0.04     1.011     0.897      0.000
 -3.00    0.000 -0.059      733      60      0.01      0.10     1.011     0.800      0.000
 -4.00    0.000 -0.064      805      65      0.00      0.15     1.059     0.600      0.000
 -5.00    0.000 -0.074      925      73      0.01      0.16     1.014     0.466      0.000
 -6.00    0.000 -0.090     1129      91      0.00      0.26     1.012     0.319      0.000
 -7.00    0.000 -0.116     1453     120      0.00      0.29     1.013     0.175      0.000
 -8.00    0.000 -0.167     2089     171      0.00      0.30     1.012     0.082      0.000
 -9.00    0.000 -0.246     3073     253      0.01      0.00     0.000 ********* **********  (LOC ASSESS ON   0)
              TRUE-SOLUTION OF TEST PACKAGE FAILED AT X =  0.00000E+00, ERROR FLAG (GLOBAL)  -3, (LOCAL)   2
           SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)
           ENDPOINT GLOBAL ERROR= 1.000E-02 *(TOL** 1.000) APPROX,      R.M.S. RESIDUAL= 0.0E+00 OVER  9 VALUES
           MAXIMUM  GLOBAL ERROR= 1.985E-02 *(TOL** 0.830) APPROX,      R.M.S. RESIDUAL= 1.4E-01 OVER  8 VALUES
           MAXIMUM DEFECT       = 0.743     *(TOL** 0.986) APPROX,      R.M.S. RESIDUAL= 4.5E-02 OVER  8 VALUES
NORMALIZED EFFICIENCY -   MAXIMUM  DEFECT
EXPECTED   EQUIV    TIME   OVHD     FCN    NO OF
ACCURACY LOG10 TOL                 CALLS   STEPS
10** -2   -1.90    0.000 -0.056      698      57
10** -3   -2.91    0.000 -0.058      729      59
10** -4   -3.93    0.000 -0.064      799      64
10** -5   -4.94    0.000 -0.073      917      72
10** -6   -5.96    0.000 -0.090     1120      90
10** -7   -6.97    0.000 -0.115     1443     119
10** -8   -7.99    0.000 -0.166     2079     170
```

C3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.030	325	26	0.01	0.06	1.017	0.846	0.000	
-2.00	0.000	-0.032	349	28	0.00	0.06	1.016	0.750	0.000	
-3.00	0.000	-0.037	397	31	0.00	0.06	1.013	0.613	0.000	
-4.00	0.000	-0.042	457	36	0.00	0.10	1.016	0.389	0.000	
-5.00	0.000	-0.052	565	44	0.01	0.31	1.012	0.205	0.000	
-6.00	0.000	-0.064	697	57	0.28	0.35	0.999	0.000	0.000	
-7.00	0.000	-0.093	1009	84	0.49	0.61	0.992	0.000	0.000	
-8.00	0.000	-0.140	1525	127	0.63	0.74	0.991	0.000	0.000	
-9.00	0.000	-0.215	2341	195	0.74	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 1.80313E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.929E-03 *(TOL** 0.705) APPROX, R.M.S. RESIDUAL= 3.8E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 2.513E-02 *(TOL** 0.812) APPROX, R.M.S. RESIDUAL= 1.3E-01 OVER 8 VALUES

MAXIMUM DEFECT = 1.04 *(TOL** 1.030) APPROX, R.M.S. RESIDUAL= 3.8E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.99	0.000	-0.030	324	25
10** -2	-1.96	0.000	-0.032	348	27

10** -3	-2.93	0.000	-0.036	393	30
10** -4	-3.90	0.000	-0.041	451	35
10** -5	-4.87	0.000	-0.051	551	42
10** -6	-5.84	0.000	-0.062	676	54
10** -7	-6.82	0.000	-0.087	951	79
10** -8	-7.79	0.000	-0.130	1415	117

C4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.077	361	28	0.01	0.06	1.017	0.857	0.000
-2.00	0.000	-0.082	385	30	0.04	0.06	1.016	0.700	0.000
-3.00	0.000	-0.093	433	33	0.38	0.41	1.013	0.576	0.000
-4.00	0.000	-0.105	493	37	4.27	4.49	1.015	0.405	0.000
-5.00	0.000	-0.128	601	45	50.60	53.26	1.015	0.156	0.000
-6.00	0.000	-0.177	829	64	307.75	308.29	0.991	0.000	0.000
-7.00	0.000	-0.259	1213	98	603.76	604.21	0.990	0.000	0.000
-8.00	0.000	-0.390	1825	152	805.87	806.74	0.990	0.000	0.000
-9.00	0.000	-0.611	2857	238	930.87	0.00	0.000	*********	********** (LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 1.80313E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 4.495E-03 *(TOL** 0.316) APPROX, R.M.S. RESIDUAL= 4.9E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 6.454E-03 *(TOL** 0.300) APPROX, R.M.S. RESIDUAL= 3.6E-01 OVER 8 VALUES

MAXIMUM DEFECT = 0.882 *(TOL** 1.000) APPROX, R.M.S. RESIDUAL= 3.0E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.95	0.000	-0.077	359	27
10** -2	-1.95	0.000	-0.082	383	29
10** -3	-2.95	0.000	-0.092	430	32
10** -4	-3.95	0.000	-0.105	489	36
10** -5	-4.95	0.000	-0.127	595	44
10** -6	-5.95	0.000	-0.175	817	63
10** -7	-6.95	0.000	-0.255	1193	96
10** -8	-7.95	0.000	-0.383	1793	149

C5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.077	49	4	0.39	0.42	0.991	0.000	0.000
-2.00	0.000	-0.077	49	4	3.88	3.89	0.991	0.000	0.000
-3.00	0.000	-0.096	61	5	4.97	4.98	0.990	0.000	0.000
-4.00	0.000	-0.153	97	7	6.74	6.77	0.990	0.000	0.000
-5.00	0.000	-0.171	109	9	8.55	8.56	0.990	0.000	0.000
-6.00	0.000	-0.247	157	13	8.65	8.66	0.990	0.000	0.000
-7.00	0.000	-0.379	241	20	7.31	7.35	0.990	0.000	0.000

-8.00	0.000	-0.549	349	29	6.94	7.01	0.990	0.000	0.000
-9.00	0.000	-0.851	541	45	6.50	6.54	0.990	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.47 *(TOL** 0.898) APPROX, R.M.S. RESIDUAL= 3.0E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 1.52 *(TOL** 0.900) APPROX, R.M.S. RESIDUAL= 2.9E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.156 *(TOL** 0.909) APPROX, R.M.S. RESIDUAL= 2.6E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.31	0.000	-0.077	49	4
10** -3	-2.41	0.000	-0.085	53	4
10** -4	-3.51	0.000	-0.125	79	6
10** -5	-4.61	0.000	-0.164	104	8
10** -6	-5.71	0.000	-0.225	143	11
10** -7	-6.81	0.000	-0.354	225	18
10** -8	-7.92	0.000	-0.534	339	28
10** -9	-9.02	0.000	-0.855	544	45

SUMMARY OVER GROUP 3

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.248	1613	128	0.39	0.42	1.019	0.836	0.000
-2.00	0.000	-0.256	1673	135	3.88	3.89	1.016	0.733	0.000

-3.00	0.000	-0.293	1841	147	4.97	4.98	1.022	0.599	0.000	
-4.00	0.000	-0.378	2153	170	6.74	6.77	1.059	0.400	0.000	
-5.00	0.000	-0.446	2645	208	50.60	53.26	1.015	0.240	0.000	
-6.00	0.000	-0.608	3461	279	307.75	308.29	1.012	0.104	0.000	
-7.00	0.000	-0.892	4901	404	603.76	604.21	1.013	0.052	0.000	
-8.00	0.000	-1.314	7289	604	805.87	806.74	1.012	0.023	0.000	
-9.00	0.000	-2.028	11141	925	930.87	6.54	0.990	0.000	0.000	(LOC ASSESS ON 239)

GROUP 4 CRK45 INT=b3

D1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.037	421	28	16.10	20.08	0.992	0.000	0.000
-2.00	0.000	-0.034	385	28	74.00	82.12	0.990	0.000	0.000
-3.00	0.000	-0.048	541	39	57.62	67.08	0.990	0.000	0.000
-4.00	0.000	-0.068	769	57	40.42	46.96	0.990	0.000	0.000
-5.00	0.000	-0.088	997	83	28.93	33.34	0.990	0.000	0.000
-6.00	0.000	-0.131	1489	124	11.61	13.30	0.990	0.000	0.000
-7.00	0.000	-0.197	2245	187	3.44	4.00	0.990	0.000	0.000
-8.00	0.000	-0.304	3457	288	15.22	17.52	0.990	0.000	0.000
-9.00	0.000	-0.474	5389	449	22.37	25.84	0.990	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

```
ENDPOINT GLOBAL ERROR=  51.4     *(TOL** 1.075) APPROX,      R.M.S. RESIDUAL= 3.3E-01 OVER  9 VALUES
MAXIMUM  GLOBAL ERROR=  60.7     *(TOL** 1.076) APPROX,      R.M.S. RESIDUAL= 3.2E-01 OVER  9 VALUES
MAXIMUM DEFECT      =   1.08     *(TOL** 1.022) APPROX,      R.M.S. RESIDUAL= 2.9E-02 OVER  9 VALUES
```

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-1.01	0.000	-0.037	420	28
10** -2	-1.99	0.000	-0.034	385	28
10** -3	-2.97	0.000	-0.047	536	38
10** -4	-3.95	0.000	-0.067	757	56
10** -5	-4.93	0.000	-0.086	980	81
10** -6	-5.91	0.000	-0.127	1442	120
10** -7	-6.88	0.000	-0.190	2157	179
10** -8	-7.86	0.000	-0.289	3290	274
10** -9	-8.84	0.000	-0.447	5082	423

D2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.039	445	27	20.64	20.64	0.995	0.000	0.000
-2.00	0.000	-0.045	517	32	28.78	63.81	0.992	0.000	0.000
-3.00	0.000	-0.063	721	44	26.29	58.35	0.992	0.000	0.000
-4.00	0.000	-0.088	997	63	35.59	75.20	0.992	0.000	0.000

-5.00	0.000	-0.123	1393	91	31.97	65.20	0.992	0.000	0.000
-6.00	0.000	-0.171	1945	132	36.86	75.07	0.991	0.000	0.000
-7.00	0.000	-0.205	2329	194	30.53	62.61	0.991	0.000	0.000
-8.00	0.000	-0.317	3601	300	3.18	6.36	0.990	0.000	0.000
-9.00	0.000	-0.495	5629	469	11.17	21.64	0.990	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 42.9 *(TOL** 1.063) APPROX, R.M.S. RESIDUAL= 2.8E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 68.7 *(TOL** 1.048) APPROX, R.M.S. RESIDUAL= 3.3E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.930 *(TOL** 1.006) APPROX, R.M.S. RESIDUAL= 4.5E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.96	0.000	-0.039	442	26
10** -2	-1.96	0.000	-0.045	513	31
10** -3	-2.95	0.000	-0.063	711	43
10** -4	-3.95	0.000	-0.086	982	61
10** -5	-4.94	0.000	-0.120	1369	89
10** -6	-5.93	0.000	-0.168	1908	129
10** -7	-6.93	0.000	-0.202	2301	189
10** -8	-7.92	0.000	-0.308	3502	291
10** -9	-8.92	0.000	-0.480	5460	454

D3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.037	421	26	5.26	21.46	1.116	0.077	0.000	
-2.00	0.000	-0.059	673	39	4.84	35.00	1.066	0.051	0.000	
-3.00	0.000	-0.074	841	53	6.08	41.81	1.039	0.075	0.000	
-4.00	0.000	-0.105	1189	75	14.51	80.52	1.004	0.040	0.000	
-5.00	0.000	-0.146	1657	107	20.24	101.38	1.004	0.009	0.000	
-6.00	0.000	-0.209	2377	154	20.86	105.53	1.003	0.006	0.000	
-7.00	0.000	-0.279	3169	223	28.67	143.93	0.998	0.000	0.000	
-8.00	0.000	-0.349	3973	331	28.41	145.10	0.996	0.000	0.000	
-9.00	0.000	-0.544	6181	515	7.18	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 2.03218E-04, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 5.25 *(TOL** 0.927) APPROX, R.M.S. RESIDUAL= 2.4E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 19.9 *(TOL** 0.879) APPROX, R.M.S. RESIDUAL= 7.6E-02 OVER 8 VALUES

MAXIMUM DEFECT = 0.924 *(TOL** 0.999) APPROX, R.M.S. RESIDUAL= 2.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.036	412	25
10** -2	-1.97	0.000	-0.058	665	38
10** -3	-2.97	0.000	-0.074	835	52
10** -4	-3.97	0.000	-0.104	1178	74

10** -5	-4.97	0.000	-0.145	1644	106
10** -6	-5.97	0.000	-0.207	2357	152
10** -7	-6.97	0.000	-0.277	3149	221
10** -8	-7.98	0.000	-0.348	3953	328

D4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.063	721	38	4.82	19.78	1.438	0.342	0.000	
-2.00	0.000	-0.081	925	51	3.41	70.07	1.310	0.294	0.000	
-3.00	0.000	-0.095	1081	67	3.30	49.36	1.397	0.164	0.000	
-4.00	0.000	-0.133	1513	95	6.71	132.70	2.468	0.211	0.000	
-5.00	0.000	-0.185	2101	134	12.05	210.56	1.070	0.157	0.000	
-6.00	0.000	-0.263	2989	192	13.71	201.68	1.029	0.156	0.000	
-7.00	0.000	-0.365	4153	277	16.71	255.02	1.035	0.155	0.000	
-8.00	0.000	-0.428	4861	402	32.11	511.92	1.049	0.107	0.000	
-9.00	0.000	-0.655	7441	620	16.90	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 6.35521E-05, ERROR FLAG (GLOBAL) -3, (LOCAL) 2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 2.53 *(TOL** 0.886) APPROX, R.M.S. RESIDUAL= 1.4E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 19.9 *(TOL** 0.825) APPROX, R.M.S. RESIDUAL= 1.3E-01 OVER 8 VALUES

MAXIMUM DEFECT = 0.883 *(TOL** 0.990) APPROX, R.M.S. RESIDUAL= 7.6E-02 OVER 8 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.96	0.000	-0.063	712	37
10** -2	-1.97	0.000	-0.081	918	50
10** -3	-2.98	0.000	-0.095	1077	66
10** -4	-3.99	0.000	-0.133	1507	94
10** -5	-5.00	0.000	-0.185	2099	133
10** -6	-6.01	0.000	-0.264	2998	192
10** -7	-7.02	0.000	-0.366	4166	279
10** -8	-8.03	0.000	-0.429	4881	405

D5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.077	877	47	17.97	19.61	1.588	0.318	0.000	(LOC ASSESS ON 44)
TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 1.89488E+01, ERROR FLAG (GLOBAL) -1, (LOCAL) 2										
-2.00	0.000	-0.130	1477	81	0.27	44.26	2.528	0.407	0.000	
-3.00	0.000	-0.162	1837	106	2.57	211.84	10.545	0.377	0.009	
-4.00	0.000	-0.205	2329	144	3.96	391.30	7.115	0.451	0.007	
-5.00	0.000	-0.279	3169	201	8.89	191.54	7.126	0.507	0.010	
-6.00	0.000	-0.389	4417	285	24.70	1718.05	1.350	0.519	0.000	
-7.00	0.000	-0.549	6241	414	31.87	2649.58	1.248	0.357	0.000	
-8.00	0.000	-0.677	7693	613	42.07	0.00	0.000	*********	**********	(LOC ASSESS ON 0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 6.78306E-05, ERROR FLAG (GLOBAL) -3, (LOCAL)   2

-9.00   0.000 -0.999   11353   946   29.13   0.00   0.000 ********* ********** (LOC ASSESS ON   0)

TRUE-SOLUTION OF TEST PACKAGE FAILED AT X = 0.00000E+00, ERROR FLAG (GLOBAL) -3, (LOCAL)   2

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.23   *(TOL** 0.826) APPROX,   R.M.S. RESIDUAL= 5.0E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 13.5   *(TOL** 0.677) APPROX,   R.M.S. RESIDUAL= 2.4E-01 OVER 6 VALUES

MAXIMUM DEFECT   = 0.996   *(TOL** 0.976) APPROX,   R.M.S. RESIDUAL= 1.8E-01 OVER 6 VALUES

NORMALIZED EFFICIENCY -   MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-2.05	0.000	-0.131	1494	82
10** -3	-3.07	0.000	-0.165	1872	108
10** -4	-4.10	0.000	-0.212	2410	149
10** -5	-5.12	0.000	-0.292	3320	211
10** -6	-6.15	0.000	-0.412	4682	303

SUMMARY OVER GROUP 4

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV	
-1.00	0.000	-0.254	2885	166	20.64	21.46	1.588	0.178	0.000	(LOC ASSESS ON 163)
-2.00	0.000	-0.350	3977	231	74.00	82.12	2.528	0.216	0.000	
-3.00	0.000	-0.442	5021	309	57.62	211.84	10.545	0.178	0.003	
-4.00	0.000	-0.598	6797	434	40.42	391.30	7.115	0.203	0.002	

-5.00	0.000	-0.820	9317	616	31.97	210.56	7.126	0.201	0.003	
-6.00	0.000	-1.163	13217	887	36.86	1718.05	1.350	0.202	0.000	
-7.00	0.000	-1.595	18137	1295	31.87	2649.58	1.248	0.147	0.000	
-8.00	0.000	-2.075	23585	1934	42.07	511.92	1.049	0.033	0.000	(LOC ASSESS ON1321)
-9.00	0.000	-3.166	35993	2999	29.13	25.84	0.990	0.000	0.000	(LOC ASSESS ON 918)

GROUP 5 CRK45 INT=b3

E1 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.006	121	9	3.02	3.02	1.024	0.222	0.000
-2.00	0.000	-0.008	157	13	3.07	3.07	1.003	0.077	0.000
-3.00	0.000	-0.012	241	20	2.06	3.14	1.030	0.050	0.000
-4.00	0.000	-0.019	361	30	2.39	3.34	1.039	0.033	0.000
-5.00	0.000	-0.028	553	46	2.56	3.54	1.056	0.022	0.000
-6.00	0.000	-0.044	865	72	2.47	3.50	1.057	0.014	0.000
-7.00	0.000	-0.069	1345	112	2.29	3.51	1.043	0.009	0.000
-8.00	0.000	-0.109	2125	177	2.58	3.51	1.024	0.006	0.000
-9.00	0.000	-0.172	3349	279	2.83	3.51	1.008	0.004	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 2.68 *(TOL** 1.004) APPROX, R.M.S. RESIDUAL= 5.2E-02 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 3.01 *(TOL** 0.991) APPROX, R.M.S. RESIDUAL= 1.2E-02 OVER 9 VALUES

MAXIMUM DEFECT = 0.975 *(TOL** 1.018) APPROX, R.M.S. RESIDUAL= 1.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.97	0.000	-0.006	119	8
10** -2	-1.95	0.000	-0.008	155	12
10** -3	-2.94	0.000	-0.012	235	19
10** -4	-3.92	0.000	-0.018	351	29
10** -5	-4.90	0.000	-0.027	534	44
10** -6	-5.88	0.000	-0.042	828	68
10** -7	-6.87	0.000	-0.066	1280	106
10** -8	-7.85	0.000	-0.103	2007	167
10** -9	-8.83	0.000	-0.161	3142	261

E2 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.026	721	39	0.23	0.54	1.298	0.308	0.000
-2.00	0.000	-0.034	937	49	0.33	0.84	8.088	0.286	0.020
-3.00	0.000	-0.044	1225	66	0.47	1.02	1.281	0.470	0.000
-4.00	0.000	-0.057	1585	90	1.30	3.15	1.277	0.322	0.000
-5.00	0.000	-0.070	1957	122	1.66	4.11	2.553	0.254	0.000
-6.00	0.000	-0.101	2797	174	1.85	4.67	4.862	0.218	0.000

-7.00	0.000	-0.138	3829	250	2.03	5.12	1.744	0.204	0.000
-8.00	0.000	-0.179	4981	365	1.82	4.61	1.524	0.189	0.000
-9.00	0.000	-0.246	6853	557	1.29	3.35	1.190	0.203	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 0.265 *(TOL** 0.889) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 0.621 *(TOL** 0.884) APPROX, R.M.S. RESIDUAL= 1.9E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.701 *(TOL** 0.972) APPROX, R.M.S. RESIDUAL= 1.0E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -2	-1.90	0.000	-0.033	915	47
10** -3	-2.93	0.000	-0.043	1204	64
10** -4	-3.96	0.000	-0.056	1569	88
10** -5	-4.98	0.000	-0.070	1951	121
10** -6	-6.01	0.000	-0.101	2811	175
10** -7	-7.04	0.000	-0.139	3877	254
10** -8	-8.07	0.000	-0.184	5114	378
10** -9	-9.10	0.000	-0.253	7040	576

E3 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.038	409	23	2.78	2.83	1.080	0.348	0.000

-2.00	0.000	-0.062	673	35	0.41	0.53	1.079	0.257	0.000
-3.00	0.000	-0.079	865	48	3.18	3.23	1.042	0.250	0.000
-4.00	0.000	-0.098	1069	66	3.69	3.77	1.105	0.364	0.000
-5.00	0.000	-0.132	1441	94	2.32	2.50	2.142	0.181	0.000
-6.00	0.000	-0.182	1981	136	1.35	1.73	1.259	0.132	0.000
-7.00	0.000	-0.246	2677	202	1.44	1.87	1.002	0.010	0.000
-8.00	0.000	-0.343	3733	302	1.73	2.22	0.998	0.000	0.000
-9.00	0.000	-0.509	5545	459	1.81	1.97	0.995	0.000	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 1.80 *(TOL** 1.000) APPROX, R.M.S. RESIDUAL= 2.7E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 1.89 *(TOL** 0.993) APPROX, R.M.S. RESIDUAL= 2.3E-01 OVER 9 VALUES

MAXIMUM DEFECT = 0.810 *(TOL** 0.989) APPROX, R.M.S. RESIDUAL= 1.9E-02 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -1	-0.92	0.000	-0.036	387	22
10** -2	-1.93	0.000	-0.060	654	34
10** -3	-2.94	0.000	-0.078	853	47
10** -4	-3.95	0.000	-0.097	1059	65
10** -5	-4.96	0.000	-0.131	1427	93
10** -6	-5.98	0.000	-0.181	1968	134
10** -7	-6.99	0.000	-0.245	2668	201
10** -8	-8.00	0.000	-0.343	3731	301

10** -9	-9.01	0.000	-0.511	5563	460

E4 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.002	49	4	0.00	0.01	1.368	0.500	0.000
-2.00	0.000	-0.002	49	4	0.01	0.05	1.368	0.500	0.000
-3.00	0.000	-0.002	49	4	0.11	0.54	1.368	0.500	0.000
-4.00	0.000	-0.003	73	5	0.08	0.39	1.368	0.400	0.000
-5.00	0.000	-0.003	85	6	0.18	0.69	1.368	0.333	0.000
-6.00	0.000	-0.005	133	8	0.29	0.57	1.368	0.375	0.000
-7.00	0.000	-0.008	193	12	0.58	0.85	1.473	0.250	0.000
-8.00	0.000	-0.011	265	17	0.80	0.88	3.711	0.176	0.000
-9.00	0.000	-0.014	349	27	0.84	1.23	1.394	0.148	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 7.438E-03 *(TOL** 0.746) APPROX, R.M.S. RESIDUAL= 2.1E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 2.931E-02 *(TOL** 0.791) APPROX, R.M.S. RESIDUAL= 3.6E-01 OVER 9 VALUES

MAXIMUM DEFECT = 2.711E-02 *(TOL** 0.796) APPROX, R.M.S. RESIDUAL= 3.5E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED ACCURACY	EQUIV LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS
10** -3	-1.80	0.000	-0.002	49	4
10** -4	-3.06	0.000	-0.002	50	4

10** -5	-4.31	0.000	-0.003	76	5
10** -6	-5.57	0.000	-0.004	112	7
10** -7	-6.82	0.000	-0.007	182	11
10** -8	-8.08	0.000	-0.011	271	17

E5 (SCALED)

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.003	49	4	0.00	0.00	2.520	1.000	0.000
-2.00	0.000	-0.003	49	4	0.01	0.01	2.520	1.000	0.000
-3.00	0.000	-0.003	49	4	0.07	0.07	2.520	1.000	0.000
-4.00	0.000	-0.004	73	5	0.08	0.08	2.800	1.000	0.000
-5.00	0.000	-0.006	97	6	0.50	0.50	1.902	1.000	0.000
-6.00	0.000	-0.007	121	7	0.93	0.93	1.538	1.000	0.000
-7.00	0.000	-0.012	193	10	1.52	1.53	1.105	1.000	0.000
-8.00	0.000	-0.018	289	14	2.25	2.28	1.048	0.929	0.000
-9.00	0.000	-0.021	349	20	2.82	2.89	1.014	0.900	0.000

SMOOTHNESS FIT OF LOG10(ERROR) VS LOG10(TOL)

ENDPOINT GLOBAL ERROR= 4.503E-03 *(TOL** 0.656) APPROX, R.M.S. RESIDUAL= 2.2E-01 OVER 9 VALUES

MAXIMUM GLOBAL ERROR= 4.542E-03 *(TOL** 0.655) APPROX, R.M.S. RESIDUAL= 2.2E-01 OVER 9 VALUES

MAXIMUM DEFECT = 2.815E-02 *(TOL** 0.796) APPROX, R.M.S. RESIDUAL= 3.5E-01 OVER 9 VALUES

NORMALIZED EFFICIENCY - MAXIMUM DEFECT

EXPECTED EQUIV TIME OVHD FCN NO OF

ACCURACY	LOG10 TOL			CALLS	STEPS
10** -3	-1.82	0.000	-0.003	49	4
10** -4	-3.08	0.000	-0.003	50	4
10** -5	-4.33	0.000	-0.005	80	5
10** -6	-5.59	0.000	-0.007	111	6
10** -7	-6.84	0.000	-0.011	181	9
10** -8	-8.10	0.000	-0.018	294	14

SUMMARY OVER GROUP 5

LOG10 TOL	TIME	OVHD	FCN CALLS	NO OF STEPS	END PNT GLB ERR	MAXIMUM GLB ERR	MAXIMUM DEFECT	FRACTION DECEIVED	FRACTION BAD DECV
-1.00	0.000	-0.075	1349	79	3.02	3.02	2.520	0.354	0.000
-2.00	0.000	-0.108	1865	105	3.07	3.07	8.088	0.286	0.010
-3.00	0.000	-0.141	2429	142	3.18	3.23	2.520	0.352	0.000
-4.00	0.000	-0.181	3161	196	3.69	3.77	2.800	0.311	0.000
-5.00	0.000	-0.240	4133	274	2.56	4.11	2.553	0.208	0.000
-6.00	0.000	-0.339	5897	397	2.47	4.67	4.862	0.169	0.000
-7.00	0.000	-0.472	8237	586	2.29	5.12	1.744	0.114	0.000
-8.00	0.000	-0.659	11393	875	2.58	4.61	3.711	0.098	0.000
-9.00	0.000	-0.962	16445	1342	2.83	3.51	1.394	0.101	0.000

SUMMARY OVER ALL GROUPS CRK45 INT=b3

LOG10	TIME	OVHD	FCN	NO OF	MAXIMUM	FRACTION	FRACTION

TOL			CALLS	STEPS	DEFECT	DECEIVED	BAD DECV	
-1.00	0.000	-0.663	7368	470	5.726	0.424	0.002	(LOC ASSESS ON 467)
-2.00	0.000	-0.839	9853	623	8.123	0.369	0.005	
-3.00	0.000	-1.041	12313	796	10.545	0.310	0.001	
-4.00	0.000	-1.373	16081	1065	7.115	0.267	0.001	
-5.00	0.000	-1.803	21541	1470	7.126	0.207	0.001	
-6.00	0.000	-2.521	30037	2095	11.487	0.172	0.000	
-7.00	0.000	-3.534	41653	3060	22.157	0.122	0.000	
-8.00	0.000	-4.867	56953	4562	32.804	0.063	0.001	(LOC ASSESS ON3949)
-9.00	0.000	-7.366	85309	7023	8.942	0.068	0.001	(LOC ASSESS ON3640)
OVERALL								
SUMMARY	0.000	-24.007	281108	21164	32.804	0.145	0.001	

Bibliography

[1] J. R. Dormand and P. J. Prince. A family of embedded Runge-Kutta formulae. *J. Comp. Appl. Math.*, 6:19–26, 1980.

[2] W. H. Enright. Analysis of error control stategies for continuous Runge-Kutta methods. *SIAM Journal on Numerical Analysis*, 26(3):588–599, 1989.

[3] W. H. Enright. A new error-control for initial value solvers. *App. Math. Comp.*, 31:288–301, May 1989.

[4] W. H. Enright. The relative efficiency of alternative defect control schemes for high-order continuous Runge-Kutta formulas. *SIAM Journal on Numerical Analysis*, 30(5):1419–1445, 1993.

[5] W. H. Enright and W. B. Hayes. Robust and reliable defect control for Runge-Kutta methods. *ACM Trans. Math. Soft.*, 2007. to appear.

[6] W. H. Enright, K. R. Jackson, S. P. Nϕrsett, and P. G. Thomsen. Interpolants for Runge-Kutta formulas. *ACM Trans. Math. Soft.*, 12(3):193–218, 1986.

[7] W. H. Enright and J. D. Pryce. Two FORTRAN packages for assessing initial value methods. *ACM Trans. Math. Soft.*, 13(1):1–27, 1987.