Publ. RIMS, Kyoto Univ.
21 (1985), 255–277

Some Implicit Fourth and Fifth Order Methods with Optimum Processes for Numerical Initial Value Problems

By

Masaharu NAKASHIMA*

§ 1. Introduction

Many areas of engineering and scientific analysis require methods for solving ordinary or partial differential equations. The progress of digital computer has significantly increased our ability to carry out the numerical solution of such equations, numerous papers have been published which deal with both the theory and practice of such solutions.

In the present paper, we concern with the numerical method of the following initial value problem of ordinary differential equation:

$$(1.1) \qquad \begin{cases} y'=f(x, y) \\ y(x_0)=y_0 . \end{cases}$$

In addition to the first-order scalar equation, it is possible to consider a system of equations or an equivalent high-order single equation. In this paper we consider only (1.1) because the numerical formulas for the system is almost same to that of scalar equation.

The discretization method for (1.1) may be classified in two categories; implicit and explicit ones. The main advantage of explicit methods is that they afford the solution explicitly at each step. However, no explicit methods are A-stable, they are inapt for stiff systems. Many authors have studied the stability problems so that several stable formulas have been proposed. The drawback for classical Runge-Kutta methods in the stability can be overcome by introducing implicit formulas.

J. C. Butcher [1] was the first who considered implicit Runge-Kutta method.

Communicated by S. Hitotumatu, June 18, 1983. Revised August 30, 1984.
* Department of Mathematics, Kagoshima University, Kagoshima 890, Japan.

The general r-stage implicit Runge-Kutta method is defined by

$$(1.2)\qquad \begin{aligned} &y_{n+1}=y_n+h\Phi(x_n, y_n; h),\\ &\Phi(x_n, y_n; h)=\sum_{i=1}^{r} w_i k_i,\\ &k_i=f(x_n+a_i h, y_n+h\sum_{j=1}^{r} b_{ij}k_j) \quad (i=1, 2,\dots, r), \end{aligned}$$

where y_n is an approximation to the exact solution $y(x_n)$ of (1.1) at the point $x_n=x_0+nh$.

After him many attempts to derive implicit Runge-Kutta method have been made. A good source of information on this topic will be found in the papers of Butcher [1], [2] and [3].

In [8] and [9], the present author has studied some explicit pseudo-Runge-Kutta method:

$$\begin{aligned} &y_{n+1}=y_{n-1}+v(y_{n-1}-y_n)+h\Phi(x_{n-1}, x_n, y_{n-1}, y_n; h),\\ &\Phi(x_{n-1}, x_n, y_{n-1}, y_n; h)=\sum_{i=1}^{r} w_{i-1}k_{i-1},\\ &k_0=f(x_{n-1}, y_{n-1}), \quad k_1=f(x_n, y_n),\\ &k_i=f(x_n+a_i h, y_n+b_i(y_n-y_{n-1})+h\sum_{j=0}^{i-1} b_{ij}k_j), \quad (i=2,\dots, r-1). \end{aligned}$$

We may replace x_{n-1}, x_n, y_{n-1} and y_n on the right-hand side with x_n, x_{n+1}, y_n and y_{n+1}, respectively, to obtain the following implicit Runge-Kutta type method:

$$(1.3)\qquad \begin{aligned} &y_{n+1}=y_n+v(y_n-y_{n+1})+h\Phi(x_n, x_{n+1}, y_n, y_{n+1}; h),\\ &\Phi(x_n, x_{n+1}, y_n, y_{n+1}; h)=\sum_{i=1}^{r} w_{i-1}k_{i-1},\\ &k_0=f(x_n, y_n), \quad k_1=f(x_{n+1}, y_{n+1}),\\ &k_i=f(x_{n+1}+a_i h, y_{n+1}+b_i(y_{n+1}-y_n)+h\sum_{j=0}^{i-1} b_{ij}k_j),\\ &\qquad (a_i\neq 0) \qquad (i=2, 3,\dots, r-1). \end{aligned}$$

We note that (1.3) is r-stage method since the value y_{n+1} is obtained by using r times functional evaluations k_i within the interval x_n and x_{n+1}. Butcher's r-stage fully implicit Runge-Kutta method of order $2r$ is all A-stable. Such method does, however, suffer a serious practical disadvantage that if an r-stage implicit Runge-Kutta method is applied to an m-dimensional system of ordinary equations, then a system of mr non-linear simultaneous algebraic equations will have to be solved exactly for the function k_i $(i=1, 2,\dots, r)$ at each step by some iterative processes. But the algorithm (1.3) reduces the effort to solve non-linear equations since there is only one m-dimensional system of equations. Next, the fully implicit Runge-Kutta method requires a suitable starting approximation of k_i ($i=1, 2,..., r$) for convergence, especially if the derivative $f(x, y)$ varies rapidly at $x=x_n$ and $y=y_n$. But our algorithm (1.3) is much easy to obtain a suitable initial approximation y_{n+1}, as will be mentioned later.

The mathematical problem of numerical integration is to give the analysis for the discretization error of numerical solution. One would be interested not only in attempting to estimate the error but also in deciding whether or not the error will grow as n increases. Thus the asymptotic behavior of error as n increases is the notion of stability. Consider a simple test problem $y'=\lambda y$, $y(x_0)=1$, which has the exact solution $y(x)=\exp\{\lambda(x-x_0)\}$. For $\lambda \in C$ and Re $\lambda<0$, we have $|y(x)| \to 0$ as x increases. Thus, it is natural that numerical solution for the above problem with fixed h tends to zero as n increases. We shall call the numerical method is A-stable if the numerical solution y_n for $y'=\lambda y$ tends to zero as $n \to \infty$ for any $\lambda \in C$ and Re $\lambda<0$. Moreover, the method is said to be L-stable if it is A-stable and, in addition, the value $|y_{n+1}/y_n|$ tends to zero as Re $\lambda \to -\infty$.

In recent years Cash [4] and Cash and Moore (Cash-Moore) [6] have proposed some methods which are closely related to (1.3). They have studied some special algorithms. Cash [4] has proposed L-stable method with $v=c_i=0$. Cash-Moore [6] has proposed A-stable and symmetric method of order 4 in the case of $v=0$ in (1.3). Based upon these results Cash-Singhal [7] has derived A-stable and symmetric methods of order 4, 6 and 8, Cash [5] has proposed some difference schemes of those methods.

We have developed some methods (1.3) which combine aspect of Newton iteration scheme with Cash's methods. We also discuss them in more details and describe of their properties.

Firstly, the author shall discuss the attainable order of the method. Secondly, we discuss the stability of the method. Thirdly the local truncation error of the method is analysed, and the choice of parameters will be also discussed. Finally some numerical tests are given. Our algorithms are superior to Cash's one in the following two points: first our algorithms increase the accuracy if we compare our algorithms with Cash's one in the same stage number, second our algorithms seem to be computationally more economical than Cash's one in solving non-linear equations. Refering to the local accuracy and the number of function evaluation per step, say the stage number, one has

$$p(3)=4,$$

and

$$p(4)=5,$$

contrary to the A-stable Cash-Moore method [6], where one only has

$$p(3)=4,$$

and to Cash [4], which is L-stable, where one has,

$$p(4)=3.$$

Here, we have put $p(r)$ to denote the order of r-stage.

§ 2. Derivation of the Method

We consider three- and four-stage methods which are obtained by setting $r=3$ and 4 in (1.3) respectively.

Throughout the paper, the coefficients are constrained by

$$a_i=c_i+\sum_{j=0}^{i-1} b_{ij}.$$

Let D be the differential operator defined by

$$D=\frac{\partial}{\partial x}+f(x_n, y_n)\frac{\partial}{\partial y}.$$

We introduce the shortened notations

$$D^i f(x_n, y_n)=T^i \ (i=1, 2,\dots, 4),\ D^i f_y(x_n, y_n)=S^i \ (i=1, 2),$$
$$(Df_y)^2(x_n, y_n)=P,\ (Df)^2(x_n, y_n)=Q,\ Df_{yy}(x_n, y_n)=R,$$
$$f_y(x_n, y_n)=f_y,\ f_{yy}(x_n, y_n)=f_{yy},$$

and we also introduce an abbreviation,

$$\Sigma=\sum_{i=2}^{3}.$$

Assume that $y_n-y(x_n)=O(h^5)$. By the Taylor series expansion about (x_n, y_n), we obtain the followings.

$$(2.1)\quad y_{n+1}=y_n+hA_1k_1+h^2A_2T+\frac{1}{2!}h^3(A_3f_yT+A_4T^2)$$
$$+\frac{1}{3!}h^4(B_1T^3+B_2f_yT^2+B_3f_y^2T+3B_4ST)$$
$$+\frac{1}{4!}h^5(C_1T^4+6C_2TS+4C_3T^2S+3C_4f_{yy}Q+C_5f_yT^3$$

$$+C_6f_y^2T+C_7f_y^3T+C_8f_yTS)+\frac{1}{5!}h^6(D_1T^5+D_2TS^3$$
$$+D_3T^2S^2+D_4T^3S+D_5f_{yy}T^2T+D_6QR+D_7TP$$
$$+D_8f_yT^4+D_9f_y^2T^3+D_{10}f_y^3T^2+D_{11}f_y^4T+D_{12}f_{yy}f_yQ$$
$$+D_{13}f_yTS^2+D_{14}f_y^2TS+D_{15}f_yT^2S)+O(h^7).$$

The constants $\{A_i\}$, $\{B_i\}$ and $\{C_i\}$ are given by

(E0) $A_1=-v+w_0+\sum_{i=1}^{3} w_i,$

(E11) $A_2=-\frac{1}{2}v-w_0+\sum \tilde{a}_iw_i,$

(E21) $A_3=-\frac{1}{3}v+w_0+\sum q_{1i}w_i,$

(E22) $A_4=-\frac{1}{3}v+w_0+\sum \tilde{a}_i^2w_i,$

(E31) $B_1=\frac{1}{4}v+w_0+\sum \tilde{a}_i^3w_i,$

(E32) $B_2=-\frac{1}{4}v-w_0+\sum q_{2i}w_i,$

(E33) $B_3=-B_2+g_1w_3,$

(E34) $B_4=-\frac{1}{4}v-w_0+\sum \tilde{a}_iq_{1i}w_i,$

(E41) $C_1=-\frac{1}{4}v+w_0+\sum \tilde{a}_i^4w_i,$

(E42) $C_2=-\frac{1}{5}v+w_0+\sum \tilde{a}_i^2q_{1i}w_i,$

(E43) $C_3=-\frac{1}{5}v+w_0+\sum \tilde{a}_iq_{2i}w_i,$

(E44) $C_4=-\frac{1}{5}v+w_0+\sum q_{1i}^2w_i,$

(E45) $C_5=-\frac{1}{5}v+w_0+\sum q_{3i}w_i,$

(E46) $C_6=C_5-\frac{1}{5}v+g_2w_3, \quad C_7=C_6,$

(E47) $C_8=-\frac{1}{5}v+3C_5+4C_3+g_3w_3,$

(E51) $D_1=-\frac{1}{6}v-w_0+\sum \tilde{a}_i^5w_i,$

(E52) $D_2=10\left(-\frac{1}{6}\bar{v}-w_0+\sum \tilde{a}_i^3q_{1i}w_i\right),$

(E53) $D_3=10\left(-\frac{1}{6}v-w_0+\sum \tilde{a}_i^2q_{2i}w_i\right),$

(E54) $D_4=5\left(-\frac{1}{6}v-w_0+\sum \tilde{a}_i q_{3i} w_i\right)$,

(E55) $D_5=10\left(-\frac{1}{6}v-w_0+\sum q_{1i} q_{2i} w_i\right)$,

(E56) $D_6=15\left(-\frac{1}{6}v-w_0+\sum a_i q_{1i} w_i\right)$,

(E57) $D_7=3(D_4+a_3 g_4 w_3)$,

(E58) $D_8=\left(-\frac{1}{6}v-w_0+\sum q_{4i} w_i\right)$,

(E59) $D_9=D_8+g_5 w_3, \quad D_{10}=D_9, \quad D_{11}=D_9$.

(E510) $D_{12}=D_5+3D_8+g_6 w_3$,

(E511) $D_{13}=D_3+6D_8+g_7 w_3$,

(E512) $D_{14}=D_4+3D_9+4D_8+g_8 w_3$,

(E513) $D_{15}=D_4+4D_8+g_9 w_3$,

where

$$p_1=c_2+1+2b_{21}, \quad p_2=c_2+1+3b_{21}, \quad p_3=c_2+1+4b_{21},$$
$$p_4=c_2+1+5b_{21}, \quad p_5=c_2+1+2b_{31}, \quad p_6=c_2+1+3b_{31},$$
$$p_7=c_2+1+4b_{31}, \quad p_8=c_2+1+5b_{31},$$
$$q_{12}=p_1, \quad q_{13}=p_5+2\tilde{a}_2 c_3, \quad q_{22}=p_2, \quad q_{23}=p_6+3\tilde{a}_2^2 c_3,$$
$$q_{32}=p_3, \quad q_{33}=p_7+4\tilde{a}_2^3 c_3, \quad q_{42}=p_4, \quad q_{43}=p_8+5\tilde{a}_2^4 c_3,$$
$$g_1=3c_3(p_1-\tilde{a}_2^2), \quad g_2=4c_3(p_2-\tilde{a}_2^3),$$
$$g_3=12c_3(\tilde{a}_2+\tilde{a}_3)(p_1-\tilde{a}_2^2), \quad g_4=20a_2 c_3(p_1-\tilde{a}_2^2),$$
$$g_5=5c_3(p_3-\tilde{a}_2^4),$$
$$g_6=15c_3((p_1+\tilde{a}_2^2)+2(p_5+2\tilde{a}_2 c_3))(p_1-\tilde{a}_2^2),$$
$$g_7=30c_3(\tilde{a}_3^2+\tilde{a}_2^2)(p_1-\tilde{a}_2^2), \quad g_8=20c_3(\tilde{a}_2+\tilde{a}_3)(p_2-\tilde{a}_2^3),$$
$$g_9=20c_3(\tilde{a}_2+\tilde{a}_3)(p_2-\tilde{a}_2^3),$$
$$\tilde{a}_i=a_i+1 \quad (i=2, 3).$$

2.1 Fifth-order Formulas with $r=4$.

To obtain fifth-order formula, the equations

(2.2) $$A_1=1, \quad A_2=\frac{1}{2}, \quad A_3=A_4=\frac{1}{3}, \quad B_i=\frac{1}{4}\ (i=1, 2, 3, 4),$$
$$C_i=\frac{1}{5}\ (i=1,\dots, 8),$$

are satisfied, which follow by equating each terms in (2.1) to the correspondings in the expansions of $y(x_{n+1})$. (See [10].) From (E32) (E33) (E45) and (E46) we have

$$p_1=\tilde{a}_2^2, \quad p_2=\tilde{a}_2^3. \tag{2.3}$$

From (E21), (E22), (E41), (E43) and (2.3) we have

$$q_{13}=\tilde{a}_3^2, \quad q_{23}=\tilde{a}_3^3. \tag{2.4}$$

We see then that the condition (2.2) can be replaced by

$$(-1)^{i-1}\frac{w_0}{(i-1)!}+\sum_{j=2}^{3}\frac{\tilde{a}_j^{i-1}}{(i-1)!}w_j=\frac{(v+1)}{i!} \qquad (i=1,\dots,5)\,, \tag{2.5}$$

$$p_1=\tilde{a}_2^2, \quad p_2=\tilde{a}_2^3, \quad q_1=\tilde{a}_3^2, \quad q_2=\tilde{a}_3^3, \tag{2.6}$$

$$-w_0+\sum q_{3i}w_i=\frac{1}{5}(v+1), \quad \sum_{i=0}^{3} w_i=v+1\,. \tag{2.7}$$

Since there are now twelve equations in fourteen unknowns, there exists two-parameter family of solutions.

The partiqular case $w_3 \neq 0$ leads to the solution

$$
\begin{aligned}
(2.8)\qquad & a_3=\frac{-(5a_2+3)}{10a_2+5}, \quad w_3=\frac{-(2a_2+1)(v+1)}{12a_3(a_3+1)(a_2-a_3)}, \\
& w_2=\frac{1}{a_2(a_2+1)}\left\{-\frac{1}{6}(v+1)-a_3(1+a_3)w_3\right\}, \\
& w_0=a_2w_2+a_3w_3+\frac{1}{2}(v+1), \\
& w_1=1-(-v+w_0+w_2+w_3), \\
& c_2=-a_2^2(2a_2+3), \\
& b_{20}=a_2^2(a_2+1), \\
& b_{21}=a_2(a_2+1)^2, \\
& b_{32}=\frac{1}{2a_2(2a_2^2+3a_2+1)}\left[\frac{1}{w_3}\left\{\frac{1}{5}(1+v)-w_0+(c_2+4b_{20})w_2\right\}\right. \\
& \qquad\qquad \left.+a_3^2(2a_3+1)\right], \\
& c_3=6a_2(a_2+1)b_{32}-3a_3^2-2a_3^3, \\
& b_{30}=-\frac{1}{2}c_3+a_2b_{32}-\frac{a_3^2}{2}, \\
& b_{31}=a_3-(c_3+b_{30}+b_{32}),
\end{aligned}
$$

provided that $a_2 \neq 0$, $a_2 \neq a_3$, $10a_2+5 \neq 0$, i.e.

$$a_2 \neq -0.6, \ -0.5, \ -0.4, \ 0.$$

2.2 Non-existence of sixth order formula with $r=4$.

We have seen that there exists four-stage method of order 5, it is natural to ask whether it is possible to increase the order in the same stage numbers. The method (1.3) is of order 6 if, in addition to (2.2),

$$D_i=\frac{1}{6}\ (i=1, 2,\dots, 15). \tag{2.9}$$

By going through the same procedure as above, we start the discussion. We now consider two cases according as w_3 is equal to zero or otherwise.

(i) The case $w_3 \neq 0$.

From (E58) (E59), (E41) and (E45) we have

$$p_3 = \tilde{a}_2^4, \tag{2.10}$$

$$q_{33}=\tilde{a}_3^4. \tag{2.11}$$

From (2.3) (2.10) and (2.11) we have

$$p_1=\tilde{a}_2^2,\quad p_2=\tilde{a}_2^3,\quad p_3=\tilde{a}_2^4,$$

which have no solution.

(ii) The case $w_3=0$.

From (2.5), (2.9) and (E51) we have

$$D_1U_1=0,\quad D_2U_2=0, \tag{2.12}$$

where

$$D_1=\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & \tilde{a}_2 & 1/2 \\ 1 & 0 & \tilde{a}_2^2 & 1/3 \\ 1 & 0 & \tilde{a}_2^3 & 1/4 \end{pmatrix},\quad U_1=\begin{pmatrix} w_1 \\ w_0 \\ w_2 \\ v+1 \end{pmatrix},$$

$$D_2=\begin{pmatrix} 1 & \tilde{a}_2^3 & 1/3 \\ 1 & \tilde{a}_2^4 & 1/4 \\ 1 & \tilde{a}_2^5 & 1/5 \end{pmatrix},\quad U_2=\begin{pmatrix} w_1 \\ w_2 \\ v+1 \end{pmatrix},$$

a simple calculation leads to

$$\det(D_1)=-\tilde{a}_2(2\tilde{a}_2-1)(\tilde{a}_2-1)/12,$$

$$\det(D_2)=\tilde{a}_2^3(5\tilde{a}_2-3)(\tilde{a}_2-1)/60.$$

The equation (2.12) is solvable if

$$\det(D_1)=0,\quad \det(D_2)=0. \tag{2.13}$$

From (2.13) we have $\tilde{a}_2=1$ which contradicts to $a_2 \neq 0$.

2.3 Fourth order formula with $r=3$.

When we may try to make fourth order method, it is required that

$$(2.14)\qquad A_1=1,\quad A_2=\frac{1}{2},\quad A_3=A_4=\frac{1}{3},\quad B_i=\frac{1}{4}\ (i=1,2,3,4).$$

As already observed in (2.2), (2.14) simplify to

$$(2.15)\qquad (-1)^{i-1}\frac{w_0}{(i-1)}+\sum\frac{\tilde{a}_j^{i-1}}{(i-1)}w_j=\frac{(v+1)}{i!}\qquad (i=1,2,\dots,5),$$
$$-w_0+\sum q_iw_i=-(v+1),$$
$$p_1=\tilde{a}_2^2.$$

There are now eight equations in seven unknowns and there exists one-parameter family of solutions. The resulting method is

$$(2.16)\qquad y_{n+1}=vy_n+(1-v)\left\{y_n+\frac{h}{6}(k_0+k_2+4k_1)\right\},$$
$$k_0=f(x_n,y_n),\quad k_1=f(x_{n+1},y_{n+1}),$$
$$k_2=f\left(x_n+\frac{h}{2},\ \frac{y_{n+1}+y_n}{2}-\frac{h}{8}(k_0-k_1)\right).$$

However, when one of w_i $(i=0,1,2)$ is equal to zero, (2.15) has no solution. For instance, let us choose $w_0=0$ and we obtain

$$(2.17)\qquad D_3U_3=0,\quad D_4U_4=0,$$
$$D_3=\begin{pmatrix}1 & 1 & 1\\ 1 & \tilde{a}_2 & 1/2\\ 1 & \tilde{a}_2^2 & 1/3\end{pmatrix},\quad U_3=\begin{pmatrix}w_1\\ w_2\\ v+1\end{pmatrix},$$
$$D_4=\begin{pmatrix}1 & \tilde{a}_2 & 1/2\\ 1 & \tilde{a}_2^2 & 1/3\\ 1 & \tilde{a}_2^3 & 1/4\end{pmatrix},\quad U_4=\begin{pmatrix}w_1\\ w_2\\ v+1\end{pmatrix}.$$

A simple calculation leads to

$$\det(D_3)=(3\tilde{a}_2-1)(\tilde{a}_2-1)/6,$$
$$\det(D_4)=\tilde{a}_2(2\tilde{a}_2-1)(\tilde{a}_2-1)/12.$$

In the same reason as in (2.13), we see that the equation (2.15) has no solution. By repeating the same analysis we can prove that for $w_1=0$, or $w_2=0$ the equation (2.15) has no solution.

2.4 No existence of order 5 with 3-stage.

Of further interest is the problem of attainable order of three stage. Let us start the discussion in a similar way as the case $r=4$.

By setting $w_3=0$ in (2.2), we have

(2.18)
$$D_5U_5=0,\quad D_6U_6=0,$$
$$D_5=\begin{pmatrix}1 & \tilde{a}_2 & 1\\ 1 & \tilde{a}_2^2 & 1/2\\ 1 & a_2^3 & 1/3\end{pmatrix},\quad U_5=\begin{pmatrix}w_1\\ w_2\\ v+1\end{pmatrix},$$
$$D_6=\begin{pmatrix}1 & \tilde{a}_2^2 & 1/2\\ 1 & \tilde{a}_2^3 & 1/3\\ 1 & \tilde{a}_2^4 & 1/4\end{pmatrix},\quad U_6=\begin{pmatrix}w_1\\ w_2\\ v+1\end{pmatrix}.$$

A simple calculation leads to

$$\det(D_5)=\tilde{a}_2(3\tilde{a}_2-1)(\tilde{a}_2-1)/6.$$
$$\det(D_6)=\det(D_4).$$

Solving $\det(D_5)=\det(D_6)=0$, we obtain $\tilde{a}_2=1$, which contradicts to $a_2\neq 0$. We therefore have the following theorem.

Theorem. *The attainable order of* 3 *and* 4 *stages is* 4 *and* 5 *respectively.*

We will make the numerical comparison of our methods with the Cash's method and Cash-Moore method. To this end, we present here the Cash's method of order 3, which is L-stable

(2.19)
$$y_{n+1}=y_n+\frac{h}{6}(k_1+2k_2+2k_3+k_0),$$
$$k_1=f(x_{n+1},\ y_{n+1}),\quad k_2=f\left(x_{n+1}-\frac{1}{2}h,\ y_{n+1}-\frac{1}{2}hk_1\right),$$
$$k_3=f\left(x_{n+1}-\frac{1}{2}h,\ y_{n+1}-\frac{1}{2}hk_2\right),\quad k_0=f(x_n,\ y_n),$$

and Cash-Moore method of order 4, which is A-stable and symmetric,

(2.20)
$$y_{n+1}=y_n+\frac{h}{6}(k_0+4k_1+k_2),$$
$$k_0=f(x_n,\ y_n),\quad k_1=f(x_{n+1},\ y_{n+1}),$$
$$k_2=f\left(x_n+\frac{h}{2},\ \frac{y_{n+1}+y_n}{2}-\frac{h}{8}(k_0-k_1)\right).$$

§ 3. Stability Analysis

We apply our method (1.3) with $r=4$ to the test equation $y'=\lambda y$. This yields

$$y_{n+1}=\frac{d_1}{d_2}y_n,$$

where

(3.1) $$d_1=(1+v)+\{w_0-c_2w_2-c_3w_3\}\bar{h}+\{(b_{20}w_2+(b_{30}-b_{32}c_2)w_3)\}\bar{h}^2+b_{32}b_{20}w_3\bar{h}^3,$$
$$d_2=(1+v)+\{w_1+(1+c_2)w_2+(1+c_3)w_3\}\bar{h}+\{b_{21}w_2+(b_{31}+b_{32}+b_{32}c_2)w_3\}\bar{h}^2+b_{32}b_{21}w_3\bar{h}^3\quad(\bar{h}=\lambda h).$$

By the definition, the formula (1.3) is A-stable if $|d_1/d_2|<1$ for all complex λ with $\mathrm{Re}(\lambda)<0$. The maximum modulus principle implies that A-stability is equivalent to the following two conditions

(i) $|d_1/d_2|\leqq 1$ for $\mathrm{Re}(\lambda)=0$,

(ii) d_1/d_2 is analytic for $\mathrm{Im}(\lambda)=0$.

From (2.8) and (3.1) we have

$$\frac{d_1}{d_2}=\frac{60(2a_2+1)+6(10a_2+4)\bar{h}-3(4a_2+1)\bar{h}^2+a_2\bar{h}^3}{60(2a_2+1)-6(10a_2+6)\bar{h}+3(4a_2+3)\bar{h}^2-(a_2+1)\bar{h}^3},$$

putting $\bar{h}=iy$ we have,

(3.1) $$|d_1^2|-|d_2^2|=\{(2u_1u_2+u_3^2-2u_1u_5-u_6^2)+(u_2^2+2u_3u_4-u_5^2-2u_6u_4)y^2-(u_4^2-a_2^2)y^2\}y^2,$$

with $u_1=60(2a_2+1)$, $u_2=3(4a_2+1)$, $u_3=6(10a_2+4)$,

$u_4=a_2+1$, $u_5=-(4a_2+3)$, $u_6=-6(10a_2+6)$.

On substituting the value u_i $(i=1, 2,\dots, 6)$ into (3.1), we obtain

$$|d_1^2|-|d_2^2|=(2a_2+1)\{720(4a_2+1)-y^4\}y^2.$$

We see that the condition (i) is satisfied if

(a) $720(4a_2+1)(2a_2+1)<0$,

(b) $2a_2+1>0$,

and the condition (ii) is also satisfied if (b) holds.

Thus we find that the method (1.3) with coefficient (2.8) is A-stable in the domain

$$D=\{(a_2, v);\ -0.50<a_2<-0.25,\ v\neq -1\}.$$

In the case of 3-stage fourth order method, stability factor d_1/d_2 is exactly the same as that in [6], then the method (1.3) with $r=3$ is A-stable.

§4. The Optimal Method

4.1 Consideration to the local error.

We define the local truncation error $T(x_n, y_n; h)$ at $x_n = x_0 + nh$ by

$$T(x_n, y_n; h) = y(x_{n+1}) - y_{n+1}.$$

If we assume that

$$|f(x, y)| \leqq M,$$

$$\left|\frac{\partial^{i+j} f(x, y)}{\partial x^i \partial y^j}\right| \leqq L^{i+j}/M^{j-1}.$$

Then the local truncation error $T(x_n, y_n; h)$ for the formula (1.3) with $r=4$ is

$$T(x_n, y_n; h) \leqq R_1 M L^5 h^6.$$

The constant R_1 in the inequality is estimated by

$$\begin{aligned}5!R_1 \leqq\; & 32|D_1| + 8|D_2| + |D_2 + 4D_8| + 4|D_3| + |2D_3 + 3D_4| + |4D_3 + 3D_4| \\ & + |2D_3 + D_4| + |D_3 + 3D_4| + 2|D_4| + |D_5 + 3D_5 + D_7| + |D_3 + D_4| \\ & + |2D_5 + 2D_7 + 3D_{15}| + |D_5 + D_7 + D_{15}| + |D_7 + 2D_{15}| + |D_{12} + D_{13}| \\ & + |D_{13} + 3D_9| + |2D_{13} + 3D_9 + D_6| + |D_{13} + D_9 + D_6| + |D_{14}| + |D_8| \\ & + |D_9| + |D_{10}| + 2|D_{13} + D_6| + |D_{13} + 2D_6| + |D_{14} + 3D_{12}| + 2|D_{11}| \\ & + 2|D_6| + |D_7| + |6D_8 + 3D_2| + |4D_8 + 3D_2| + |D_8 + D_2| + |2D_{10} + D_{14}| \\ & + |D_{10} + D_{12} + D_{14}| + |D_{15} + D_5| + |2D_3 + 3D_4| + |D_{15}|.\end{aligned}$$

Let us denote the expression on the right hand side as $m(a_2, v)$. The error bound $m(a_2, v)$ is shown in Figure (1). We see that $m(a_2, v)$ is minimized if we set $v = -1$ which implies $R_1 = 0$. Unfortunately it is impossible to take $v = -1$. If we set, for example, $v = 0.0$ and $a_2 = -0.25 \sim -0.35$, then we have

$$R_1 = 0.03 \sim 0.05.$$

The local truncation error for the formula (2.16) is

$$T(x_n, y_n; h) \leqq R_2 M L^4 h^5,$$

where the constant R_2 is estimated by

$$\begin{aligned}4!R_2 \leqq\; & 16|C_1| + 24|C_2| + |6C_2 + 3C_5| + |12C_2 + 3C_5| + |6C_2 + C_5| \\ & + |C_5| + 32|C_3| + |2C_6 + C_8| + |C_6 + 3C_4 + C_8| + |3C_4|.\end{aligned}$$

Let us denote the expression on the right hand side as $m(v)$, the error bound

$m(v)$ is shown in Figure (2).

If we set $v=-0.07\sim-0.08$ in (2.16), then we have

$$R_2=0.02\sim 0.08.$$

The error bound for Cash-Moore method (2.20) is

$$R_2=0.05.$$

4.2 Consideration to the iteration.

We first observe that the formulae (1.3) may be rewritten as

$$y_{n+1}=y_n+(1+v)\{y_n+h\tilde{\Phi}(x_n, x_{n+1}, y_n, y_{n+1}; h)-y_{n+1}\},$$

where

$$\tilde{\Phi}(x_n, x_{n+1}, y_n, y_{n+1}; h)=\sum_{i=1}^{r} \tilde{w}_{i-1}k_{i-1}.$$

In this equation $\tilde{w}_i$ $(i=0, 1, 2, 3)$ are equal to w_i $(i=0, 1, 2, 3)$ by imposing $v=0$ in (2.8), respectively. In general, this form is non-linear equations, therefore we must solve the equation by some iterative processes. Natural is the successive substitution procedure.

$$(4.1)\qquad y_{n+1}^{(s+1)}=y_{n+1}^{(s)}+(1+v)\{y_n+h\tilde{\Phi}(x_n, x_{n+1}, y_n, y_{n+1}^{(s)}; h)-y_{n+1}^{(s)}\},$$

where the superscript indicates iteration number. From the stand point of economic computation, the rate of convergence to the solution is important. Using the mean value theorem, we have

$$\begin{aligned} y_{n+1}^{(s+1)}-y_{n+1}^{(s)} &= \{1-(1+v)+(1+v)h\tilde{\Phi}_v(x_n, x_{n+1}, y_n, \xi_{n+1}^{(s)}; h)\}(y_{n+1}^{(s)}-y_{n+1}^{(s-1)}) \\ &= r^{(s)}(y_{n+1}^{(s)}-y_{n+1}^{(s-1)}), \end{aligned}$$

where

$$\tilde{\Phi}_v=\frac{\partial\tilde{\Phi}}{\partial v}(x, y, u, v),$$

$$\xi_{(n+1)}^{(s)}=y_{n+1}^{(s)}+\theta_{n+1}^{(s)}(y_{n+1}^{(s)}-y_{n+1}^{(s-1)}),\quad 0\leqq\theta_{n+1}^{(s)}\leqq 1.$$

In order that the iteration (4.1) converges for all $y_{n+1}^{(1)}$, it is necessary that $|r^{(s)}|<1$. If the value of $\tilde{\Phi}_v$ is large, we may choose h so that $|(1+v)h\tilde{\Phi}_v|<1$ and $|r^{(s)}|=|-v+(1+v)h\tilde{\Phi}_v|<1$. Clearly, the smaller is v, the faster is the rate of convergence of the processes (4.1).

Thus, for stiff problems, the optimum value of v lies near zero.

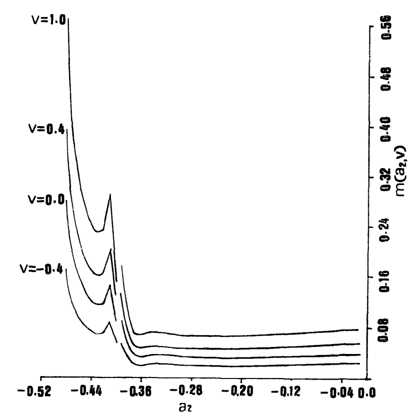

Figure 1. The error bound for the method (2.1) with the coefficients (2.8) in the domain $D=\{(a_2, v);\ -0.5 < a_2 < -0.25,\ |v| \leqq 1\}$.

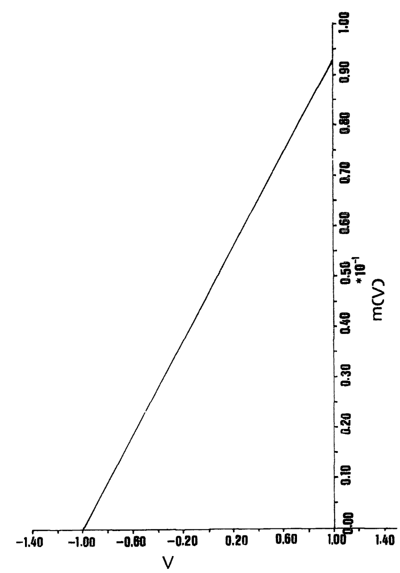

Figure 2. The error bound for the method (2.15) in the domin $D=\{v;\ |v| \leqq 1\}$.

§ 5. Numerical Example

In this section, we present some numerical results for the equations which have been often taken up in the literature of the numerical analysis.

We use the following initial-value problems.

$$1:\quad \begin{cases} y' = -5y+4z, & y(0)=-3, \quad y(x)=\exp(-x)-4\exp(-10x), \\ z' = 5y-6z, & z(0)=6, \quad\ \ z(x)=\exp(-x)+5\exp(-10x), \end{cases}$$

$$11:\quad \begin{cases} y' = -0.01y+1000z, & y(0)=\dfrac{499.99}{1499.99}, \\ z' = -1500z, & z(0)=1, \end{cases}$$

$$y(x)=\exp(-0.01x)-\frac{1000}{1499.99}\exp(-1500x)$$

$$z(x)=\exp(-1500x),$$

$$111:\quad \begin{cases} y'=0.01-(0.01+y+z)[1+(y+1000)(y+1)], & y(0)=0, \\ z'=0.01-(0.01+y+z)(1+z^2), & z(0)=0. \end{cases}$$

The problem 111 is non-linear stiff, whose Jacobian has the eigenvalues -1012 and -0.089 at $x=0$ and -21.7 and -0.089 at $x=100$. Since it has no analytical solution, we compute an exact solution using the fourth order explicit Runge-Kutta methods with very fine step-size. To keep the A-stability of the fourth order Runge-Kutta method in $0 \leqq x \leqq 100$, we take the step size to be 5×10^{-4}.

We set the initial approximation $y_{n+1}^{(1)}$ on the iterative processes (4.1) by

$$y_{n+1}^{(1)} = y_n + hk_0 .$$

We use the quantity

$$\varepsilon_n^{(i)} = \left| |y_{n+1}^{(i+1)}| - |y_{n+1}^{(i)}| \right| ,$$

as an control of the iteration number. The iteration is continued until $\varepsilon_n^{(i)}$ become smaller than E, where E is a pre-assigned tolerance. From Tables it can be seen that the advantage of our methods lies in the following points. Firstly comparing both methods in the same stage number, our algorithms are more accurate than Cash's one, lastly, we see that the convergence of the iteration of our algorithms are faster than Cash's one, especially when the value y_n varies rapidly. From those results, our algorithms are more efficient than Cash's one.

Computations were done in double precision arithmetic on the FACOM M-200 of Kyushu University.

The followings are comparison of the error incurred by using the Cash's methods (2.16), (2.20) and the methods (2.8), (2.16).

Table 1 (1)

Problem 1, $h=1/2^5$, $E=10^{-7}$, M: number of iterations.

Absolute Error

	Cash's method (2.20)			Method (2.1) with (2.16) ($v=-0.09$)		
x	$y_n-y(x_n)$	$z_n-z(x_n)$	M	$y_n-y(x_n)$	$z_n-z(x_n)$	M
0.0625	0.182E−4	−0.230E−4	9	0.182E−4	−0.230E−4	7
0.1875	0.153E−4	−0.202E−4	9	0.153E−4	−0.202E−4	6
0.3125	0.724E−5	−0.938E−5	8	0.727E−5	−0.938E−5	6
0.5000	0.165E−5	−0.247E−5	7	0.166E−5	−0.245E−5	5
0.7500	0.579E−7	−0.417E−6	6	0.871E−7	−0.415E−6	5
1.0000	−0.804E−7	−0.122E−6	4	−0.209E−7	−0.843E−7	4
1.5000	−0.891E−7	−0.108E−6	3	0.335E−9	−0.111E−7	4
2.0000	−0.680E−7	−0.859E−7	3	0.272E−7	0.934E−8	4

	Cash's method (2.19)			Method (2.1) with (2.8) ($a_2=-0.35$, $v=-0.09$)		
x	$y_n-y(x_n)$	$z_n-z(x_n)$	M	$y_n-y(x_n)$	$z_n-z(x_n)$	M
0.0625	0.648E 0	0.138E 0	31	0.230E−5	−0.287E−5	10
0.1875	0.102E 0	0.503E 0	13	0.170E−5	−0.271E−5	9
0.3125	0.986E 0	0.688E 0	12	0.801E−6	−0.928E−6	8
0.5000	0.796E 0	0.713E 0	11	0.177E−6	−0.284E−6	7
0.7500	0.533E−5	−0.724E−5	8	0.234E−7	−0.484E−7	6
1.0000	0.385E−6	−0.970E−6	6	0.111E−7	−0.267E−7	5
1.5000	−0.202E−6	−0.226E−6	3	0.304E−8	−0.619E−8	4
2.0000	−0.181E−6	−0.200E−6	3	0.160E−7	−0.285E−8	4

Table 1 (2)

Relative Error

x	Cash's method (2.19) $\left\lvert\frac{y_n-y(x_n)}{y(x_n)}\right\rvert$	Cash's method (2.19) $\left\lvert\frac{z_n-z(x_n)}{z(x_n)}\right\rvert$	Cash's method (2.20) $\left\lvert\frac{y_n-y(x_n)}{y(x_n)}\right\rvert$	Cash's method (2.20) $\left\lvert\frac{z_n-z(x_n)}{z(x_n)}\right\rvert$	Method (2.1) with (2.8), ($a_2=-0.35$, $v=-0.09$) $\left\lvert\frac{y_n-y(x_n)}{y(x_n)}\right\rvert$	Method (2.1) with (2.8), ($a_2=-0.35$, $v=-0.09$) $\left\lvert\frac{z_n-z(x_n)}{z(x_n)}\right\rvert$	Method (2.1) with (2.16) ($v=-0.09$) $\left\lvert\frac{y_n-y(x_n)}{y(x_n)}\right\rvert$	Method (2.1) with (2.16) ($v=-0.09$) $\left\lvert\frac{z_n-z(x_n)}{z(x_n)}\right\rvert$
0.0625	0.569E+0	0.384E−1	0.152E−4	0.637E−5	0.191E−5	0.796E−6	0.152E−4	0.637E−5
0.1875	0.476E+1	0.315E+0	0.713E−4	0.126E−4	0.791E−5	0.170E−5	0.712E−4	0.126E−4
0.3125	0.174E+1	0.723E+0	0.130E−4	0.986E−5	0.144E−5	0.976E−6	0.130E−4	0.986E−5
0.5000	0.137E+1	0.111E+1	0.285E−5	0.386E−5	0.306E−6	0.443E−6	0.287E−5	0.383E−5
0.7500	0.113E−4	0.152E−4	0.123E−6	0.879E−6	0.498E−7	0.101E−6	0.185E−6	0.874E−6
1.0000	0.104E−5	0.263E−5	0.218E−6	0.333E−6	0.303E−7	0.725E−7	0.569E−7	0.299E−6
1.5000	0.908E−6	0.101E−5	0.399E−6	0.488E−6	0.136E−7	0.277E−7	0.150E−8	0.500E−7
2.0000	0.133E−5	0.147E−5	0.502E−6	0.635E−6	0.118E−6	0.211E−7	0.201E−6	0.690E−7

Computational time

Cash's methods		Method (2.1)	
(2.19)	(2.20)	(2.8)	(2.16)
0.47 (sec)	0.38 (sec)	0.44 (sec)	0.38 (sec)

Table 2 (1)

Problem 11, $h=1/2^{11}$, $E=10^{-7}$, M: number of iterations.

Absolute Error

x	Cash's method (2.20) $y_n-y(x_n)$	Cash's method (2.20) $z_n-z(x_n)$	M	Method (2.1) with (2.16) ($v=-0.09$) $y_n-y(x_n)$	Method (2.1) with (2.16) ($v=-0.09$) $z_n-z(x_n)$	M
$2/2^{11}$	0.595E−5	−0.896E−5	10	0.594E−5	−0.896E−5	7
$4/2^{11}$	0.576E−5	−0.862E−5	9	0.576E−5	−0.861E−5	7
$8/2^{11}$	0.262E−5	−0.395E−5	8	0.262E−5	−0.396E−5	6
$14/2^{11}$	0.531E−6	−0.789E−6	7	0.515E−6	−0.782E−6	5
$20/2^{11}$	0.786E−7	−0.128E−6	5	0.582E−7	−0.114E−6	4
1.0	−0.164E−7	0	2	−0.275E−7	0	2
2.0	−0.158E−7	0	2	−0.268E−7	0	2
5.0	−0.192E−8	0	2	−0.122E−7	0	2
10.0	−0.295E−7	0	2	−0.389E−7	0	2
20.0	−0.421E−7	0	2	−0.497E−7	0	2

x	Cash's method (2.19) $y_n-y(x_n)$	Cash's method (2.19) $z_n-z(x_n)$	M	Method (2.1) with (2.8) ($a_2=-0.35$, $v=-0.09$) $y_n-y(x_n)$	Method (2.1) with (2.8) ($a_2=-0.35$, $v=-0.09$) $z_n-z(x_n)$	M
$2/2^{11}$	0.122E−3	−0.183E−3	14	0.685E−6	−0.103E−5	10
$4/2^{11}$	0.117E−3	−0.176E−3	14	0.696E−6	−0.987E−6	10
$8/2^{11}$	0.545E−4	−0.817E−4	12	0.317E−6	−0.473E−6	9
$14/2^{11}$	0.106E−4	−0.158E−4	10	0.869E−7	−0.922E−7	7
$20/2^{11}$	0.170E−5	−0.251E−5	8	0.123E−7	0.124E−8	6
1.0	0.207E−7	0	2	0.465E−8	0	2
2.0	0.208E−7	0	2	0.606E−8	0	2
5.0	0.337E−7	0	2	0.225E−7	0	2
10.0	0.440E−8	0	2	−0.135E−8	0	2
20.0	−0.113E−7	0	2	−0.755E−8	0	2

Table 2 (2)

Relative Error

	Cash's method				Method (2.1)			
	(2.19)		(2.20)		with (2.8), ($a_2 = -0.35$, $\nu = -0.09$)		with (2.16) ($\nu = -0.09$)	
x	$\left\|\frac{y_n - y(x_n)}{y(x_n)}\right\|$	$\left\|\frac{z_n - z(x_n)}{z(x_n)}\right\|$	$\left\|\frac{y_n - y(x_n)}{y(x_n)}\right\|$	$\left\|\frac{z_n - z(x_n)}{z(x_n)}\right\|$	$\left\|\frac{y_n - y(x_n)}{y(x_n)}\right\|$	$\left\|\frac{z_n - z(x_n)}{z(x_n)}\right\|$	$\left\|\frac{y_n - y(x_n)}{y(x_n)}\right\|$	$\left\|\frac{z_n - z(x_n)}{z(x_n)}\right\|$
$2/2^{11}$	0.180E−3	0.382E−3	0.875E−5	0.186E−4	0.100E−5	0.214E−5	0.875E−5	0.186E−4
$4/2^{11}$	0.196E−3	0.765E−3	0.681E−5	0.373E−4	0.823E−6	0.427E−5	0.681E−5	0.372E−4
$8/2^{11}$	0.565E−4	0.153E−2	0.272E−5	0.741E−4	0.329E−6	0.886E−5	0.271E−5	0.742E−4
$14/2^{11}$	0.106E−4	0.267E−2	0.533E−6	0.133E−3	0.873E−7	0.155E−4	0.517E−6	0.131E−3
$20/2^{11}$	0.170E−5	0.380E−2	0.787E−7	0.195E−3	0.123E−7	0.188E−5	0.582E−7	0.174E−3
1.00	0.209E−7	0	0.166E−7	0	0.469E−8	0	0.278E−7	0
2.00	0.213E−7	0	0.162E−7	0	0.618E−8	0	0.273E−7	0
5.00	0.355E−7	0	0.202E−8	0	0.236E−7	0	0.128E−7	0
10.0	0.486E−8	0	0.326E−7	0	0.149E−8	0	0.430E−7	0
20.0	0.138E−7	0	0.514E−7	0	0.922E−8	0	0.607E−7	0

Computational time

Cash's methods		Method (2.1)	
(2.19)	(2.20)	(2.8)	(2.16)
2.37 (sec)	2.84 (sec)	2.37 (sec)	2.69 (sec)

Table 3 (1)

Problem 111, $h=5/10^4$, $E=10^{-7}$, M: number of iterations.
Explicit Runge-Kutta Method (order 4)

x	$\hat{y}_n$	$\hat{z}_n$
0.001	−0.199E−1	0.996E−2
0.002	−0.299E−1	0.199E−1
0.004	−0.498E−1	0.399E−1
0.007	−0.798E−1	0.698E−1
0.010	−0.109E+0	0.997E−1
1.0	−0.199E−1	0.996E−2
2.0	−0.299E−1	0.199E−1
10.0	−0.109E+0	0.997E−1
20.0	−0.209E+0	0.199E+0
40.0	−0.408E+0	0.398E+0
100.0	−0.991E+0	0.983E+0

Table 3 (2)

Absolute Error

	Cash's method (2.20)			Method (2.1) with (2.16)		
x	$y_n-\hat{y}_n$	$z_n-\hat{z}_n$	M	$y_n-\hat{y}_n$	$z_n-\hat{z}_n$	M
0.001	−0.136E− 1	0.992E− 2	6	−0.136E− 5	0.996E− 2	6
0.002	−0.212E− 1	0.199E− 1	6	−0.212E− 1	0.199E− 1	7
0.004	−0.400E− 1	0.398E− 1	5	−0.400E− 1	0.398E− 1	5
0.007	−0.697E− 1	0.697E− 1	4	−0.697E− 1	0.697E− 1	5
0.010	−0.996E− 1	0.996E− 1	2	−0.996E− 1	0.996E− 1	2
1.0	−0.766E−10	0.761E−10	2	−0.808E−10	0.812E−10	2
2.0	−0.751E−10	0.761E−10	2	−0.810E−10	0.807E−10	2
10.0	−0.726E−10	0.736E−10	2	−0.754E−10	0.749E−10	2
20.0	−0.732E−10	0.736E−10	2	−0.677E−10	0.683E−10	2
40.0	−0.701E−10	0.699E−10	2	−0.408E−10	0.406E−10	2
100.0	−0.260E−10	0.314E−10	2	0.495E− 8	−0.554E− 8	2

Table 3 (2)

	Cash's method (2.19)			Method (2.1) with (2.8) ($a_2=-0.35$, $v=-0.09$)		
x	$y_n-\hat{y}_n$	$z_n-\hat{z}_n$	M	$y_n-\hat{y}_n$	$z_n-\hat{z}_n$	M
0.001	−0.143E−1	0.996E−2	17	−0.136E−1	0.996E−2	10
0.002	−0.218E−1	0.199E−1	14	−0.213E−1	0.199E−1	9
0.004	−0.402E−1	0.398E−1	11	−0.400E−1	0.398E−1	7
0.007	−0.698E−1	0.697E−1	9	−0.697E−1	0.697E−1	5
0.010	−0.996E−1	0.996E−1	7	−0.996E−1	0.996E−1	3
1.0	−0.961E−5	0.636E−5	6	−0.616E−6	0.432E−6	2
2.0	−0.136E−4	0.103E−4	6	−0.881E−6	0.686E−6	2
10.0	−0.475E−4	0.436E−4	5	−0.311E−5	0.289E−5	2
20.0	−0.953E−4	0.910E−4	5	−0.627E−5	0.601E−5	2
40.0	−0.220E−3	0.214E−3	4	−0.145E−4	0.141E−4	2
100.0	−0.205E−2	0.210E−2	2	−0.130E−3	0.136E−3	2

Table 3 (3)

Relative Error

	Cash's method				Method (2.1)			
	(2.19)		(2.20)		with (2.8), ($a_2=-0.35$, $v=-0.09$)		with (2.16) ($v=-0.09$)	
x	$\left\lvert\frac{y_n-\hat{y}_n}{\hat{y}_n}\right\rvert$	$\left\lvert\frac{z_n-\hat{z}_n}{\hat{z}_n}\right\rvert$	$\left\lvert\frac{y_n-\hat{y}_n}{\hat{y}_n}\right\rvert$	$\left\lvert\frac{z_n-\hat{z}_n}{\hat{z}_n}\right\rvert$	$\left\lvert\frac{y_n-\hat{y}_n}{\hat{y}_n}\right\rvert$	$\left\lvert\frac{z_n-\hat{z}_n}{\hat{z}_n}\right\rvert$	$\left\lvert\frac{y_n-\hat{y}_n}{\hat{y}_n}\right\rvert$	$\left\lvert\frac{z_n-\hat{z}_n}{\hat{z}_n}\right\rvert$
0.001	0.719E+0	0.992E+0	0.686E+0	0.999E+0	0.683E+0	0.996E+0	0.683E+0	0.999E+0
0.002	0.730E+0	0.999E+0	0.712E+0	0.999E+0	0.711E+0	0.993E+0	0.711E+0	0.999E+0
0.004	0.806E+0	0.999E+0	0.803E+0	0.999E+0	0.803E+0	0.999E+0	0.803E+0	0.999E+0
0.007	0.874E+0	0.999E+0	0.874E+0	0.999E+0	0.874E+0	0.999E+0	0.874E+0	0.999E+0
0.010	0.908E+0	0.999E+0	0.908E+0	0.425E−4	0.908E+0	0.999E+0	0.908E+0	0.999E+0
1.0	0.428E−3	0.638E−3	0.308E−4	0.425E−4	0.384E−8	0.736E−8	0.405E−8	0.814E−8
2.0	0.454E−3	0.517E−3	0.251E−8	0.381E−8	0.299E−4	0.301E−4	0.270E−8	0.404E−8
10.0	0.432E−3	0.437E−3	0.661E−9	0.738E−9	0.284E−4	0.290E−4	0.687E−9	0.751E−9
20.0	0.455E−3	0.456E−3	0.349E−9	0.369E−9	0.299E−4	0.301E−4	0.323E−9	0.342E−9
40.0	0.539E−3	0.538E−3	0.171E−9	0.175E−9	0.355E−4	0.355E−4	0.998E−10	0.101E−9
100.0	0.206E−2	0.214E−2	0.262E−10	0.319E−10	0.131E−3	0.138E−3	0.499E−8	0.563E−8

Computational time

Cash's methods		Method (2.1)	
(2.19)	(2.20)	(2.8)	(2.16)
20.52 (sec)	11.21 (sec)	13.51 (sec)	10.94 (sec)

Acknowledgements

The author wishes to express his hearty thanks to Prof. S. Hitotumatu of Kyoto University, Prof. M. Tanaka of Yamanashi University and Prof. T. Mitsui of Fukui University for their invaluable suggestions and advice. The author also wishes to thank the referees for invaluable guidance and criticisms which have greatly improved the original draft.

References

[1] Butcher, J. C., Coefficients for the study of Runge-Kutta integration processes, *J. Austral. Math. Soc.* **3** (1963) 185–201.

[2] ———, Implicit Runge-Kutta processes, *Math. Comp.* **18** (1964) 50–64.

[3] ———, Integration process based on Radau quadrature formulas, *Math. Comp.* **18** (1964) 233–234.

[4] Cash, J. R., A class of implicit Runge-Kutta methods for the numerical integration of stiff ordinary differential equations, *J. ACM.* **22** (1975), 504–511.

[5] ———, On the solution of block tridiagonal systems of linear algebraic equations having a special structure, *SIAM J. Numer. Anal.* **19** (1982) 1220–1232.

[6] Cash, J. R., & Moore, D. R., A high order method for the numerical solution of two-point boundary value problems, *BIT.* **20** (1979) 44–52.

[7] Cash, J. R. and Singhal, A., High order methods for the numerical solution of two-point boundary value problems, *BIT.* **22** (1982), 184–190.

[8] Nakashima, M., On Pseudo-Runge-Kutta methods with 2 and 3 stages, *Publ. RIMS, Kyoto Univ* **18** (1982) 895–909.

[9] ———, On a Pseudo-Runge-Kutta method of order 6, *Proc. Japan Acad.*, **58** (1982) 66–68.

[10] ———, Implicit Pseudo-Runge-Kutta processes, *Publ. RIMS, Kyoto Univ* **20** (1984) 39–56.

[11] Mitsui, T., Runge-Kutta type integration formulas including the evaluation of the second derivative. Part I, *Publ. RIMS, Kyoto Univ.* **18** (1982) 325–364.

[12] Tanaka, M., Pseudo-Runge-Kutta methods and their application to the estimation of truncation errors in 2nd and 3rd order Runge-Kutta methods, *Joho Shori.* **6** (1969), 406–417 (in Japanese).