Report No. UIUCDCS-R-81-1069

MODIFIED RUNGE-KUTTA METHODS FOR SOLVING ODES

by

Thu Van Vu

DISCLAIMER

[illegible]

September 1981

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
URBANA, ILLINOIS 61801

Supported in part by the Department of Energy grant DOE DEAC02 76ER02383, and submitted in partial fulfillment of the requirements of the Graduate College for the degree of Master of Science.

ACKNOWLEDGEMENT

I wish to thank Professor C. W. Gear for his suggestion of the topic, his guidance and support during the academic year 1980-1981 and during the writing of this thesis. Also, I would like to thank all the persons working in rooms 31a-f for their friendship and their assistance in using various computer systems, as well as Mrs. Barbara Armstrong for her help in typing the thesis. Finally, special thanks are reserved for my wife Lanh for her love and encouragement throughout these years.

TABLE OF CONTENTS

Page

1. INTRODUCTION 1

2. ERROR ESTIMATION 5

3. REGIONS OF ABSOLUTE STABILITY 13

4. NUMERICAL IMPLEMENTATION 28

LIST OF REFERENCES 36

1. INTRODUCTION

Consider the initial value problem for a system of ordinary differential equations

$$y'(t) = f(t,y), \qquad y(0) = y_0. \tag{1.1}$$

To solve (1.1) numerically, a conventional q-stage pth-order Runge-Kutta (RK) method proceeds from t_n to $t_{n+1} = t_n + h$ by evaluating

$$k_i = hf(t_n + h\alpha_i\ , y_n + \sum_{j=1}^{i} \beta_{ij} k_{j-1}), \quad i = 0,\dots,q-1 \tag{1.2a}$$

where $\alpha_i = \sum_{j=1}^{i} \beta_{ij}$, and combining the values k_i to yield

$$y_{n+1} = y_n + \sum_{i=1}^{q} k_{i-1}\gamma_i. \tag{1.2b}$$

The coefficients β_{ij} and γ_i are determined so that when the true solution of (1.1) at t_n, $y(t_n)$ is substitnted into (1.2) and (1.3), the value y_{n+1} should agree with the Taylor expansion of $y(t_{n+1})$ at t_n,

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2!} y^{(2)}(t_n) + \cdots + \frac{h^p}{p!} y^{(p)}(t_n) + \cdots$$

in at least the first p + 1 terms. This pth-order accuracy is usually denoted as

$$y_{n+1} = y(t_{n+1}) + O(h^{p+1}). \tag{1.3}$$

Solving (1.1) with a Runge-Kutta method as described above is rather expensive since it requires at least q function evaluations at any t_n. First, the k_i's are obtained to form y_{n+1}. Then, if based on some kind of error estimate, it is decided that y_{n+1} is not accurate enough, the current stepsize h is reduced and the k_i's are re-evaluated to obtain another value for y_{n+1}. This last step is repeated until y_{n+1} succeeds the error test. So, the number of function evaluations is $q + m(q-1)$, where m is non-negative but not always 0. Moreover, in order to keep m low, h is usually chosen in a somewhat conservative manner, to a degree that the method is not as efficient as it should be.

Fortunately, there is a way to avoid the cost of re-evaluation. In developing an RK-like technique for starting an automatic high order multistep ODE integrator, Gear [3] proved the existence of values β_{ij}, $i=0,\ldots,q'-1$, $j=1,\ldots,i$ and γ_{js}, $j=1,\ldots q'$, $s=1,\ldots,p$ which, when used to compute k_i from (1.2a), give pth-order accurate approximations of the first p scaled derivatives $h^s y^{(s)}(t_n)/s!$, namely

$$\frac{h^s}{s!}\, y^{(s)}(t_n) = \sum_{j=1}^{q'} k_{j-1}\gamma_{js} + O(h^{p+1}), \qquad s = 1,\ldots,p. \tag{1.4}$$

Since a Taylor's series method of order p takes the form

$$y_{n+1} = y_n + hy'(t_n) + \frac{h^2}{2!}\, y^{(2)}(t_n) + \cdots + \frac{h^p}{p!}\, y^{(p)}(t_n), \tag{1.5}$$

(1.4) naturally suggests the formulation of the following modified q'-stage pth-order Runge-Kutta method:

(1) Choose a stepsize h_R. Calculate k_i and the scaled derivatives based on h_R,

$$k_i = h_R f(t_n + h_R \alpha_i\ ,\ y_n + \sum_{j=1}^{i} \beta_{ij} k_{j-1}), \qquad i = 0,\ldots,q'-1, \qquad (1.6a)$$

$$\frac{h_R^s}{s!}\, y^{(s)}(t_n) = \sum_{j=1}^{q'} k_{j-1}\gamma_{js} + O(h_R^{p+1}), \qquad s = 1,\ldots,p. \qquad (1.6b)$$

(2) Estimate the adjusted stepsize h based on an error estimate and scale the derivatives in (1.6b) accordingly,

$$\frac{h^s}{s!}\, y^{(s)}(t_n) = r^s \frac{h_R^s}{s!}\, y^{(s)}(t_n)$$

$$= r^s \sum_{j=1}^{q'} k_{j-1}\gamma_{js} + O(h^{p+1}), \qquad s = 1,\ldots,p \qquad (1.6c)$$

where $r = h/h_R$, $r_{min} < r < r_{max}$ for some r_{min}, r_{max}, and use (1.5) to obtain y_{n+1} which now can be expressed as

$$y_{n+1} = y_n + \sum_{s=1}^{p} r^s \sum_{j=1}^{q'} k_{j-1}\gamma_{js}. \qquad (1.6d)$$

y_{n+1} obtained in this way also satisfies (1.3), therefore is accurate to a pth-order. Observe that h in (1.6c) must be determined so as to yield an appropriately small estimate of the $O(h^{p+1})$ term. Once a reliable error estimator has been devised, this can be done with no substantial efforts since there is no need to re-evaluate k_i. Certainly, some conditions must be imposed on the ratio h/h_R, but if the stepsizes are controlled properly, the modified Runge-Kutta method can be superior to a conventional one.

In this thesis, questions raised by the proposed method are studied. The questions considered are:

1. How to estimate the local truncation error and accordingly adjust the stepsize h?

2. What is the region of absolute stability which is now a function of the ratio $r = h/h_R$? What is the value of r that yields the largest region?

3. How does the modified Runge-Kutta method compare to a conventional one in numerical testing?

For practical purposes, we limit ourselves to methods of fourth order. Nevertheless, o . could hope to get a general understanding of this class of methods without the necessity of going into the overwhelming mathematics of methods of order greater than four.

2. ERROR ESTIMATION

In this section, we discuss in details as how to construct an error estimator required by the algorithm (1.6a-d). What we really need to estimate here is the local truncation error of $\bar{y}_{n+1}$ defined by

$$\bar{y}_{n+1} = y_n + \sum_{s=1}^{p} \sum_{j=1}^{q'} k_{j-1}\gamma_{js},$$

and accurate to a pth-order, that is,

$$\bar{y}_{n+1} = y(t_{n+1}) + O(h_R^{p+1}).$$

For this purpose, we employ a technique similar to the one used in the Fehlberg formulas (see Fehlberg [2], Bettis [1].) Our objective is to find $\hat{y}_{n+1}$ which is also some combination of k_i, but is of one order higher in accuracy,

$$\hat{y}_{n+1} = y_n + \sum_{j=1}^{q'} k_{j-1}\hat{\gamma}_j,$$
$$\hat{y}_{n+1} = y(t_{n+1}) + O(h_R^{p+2}),$$

and to use $\hat{y}_{n+1} - \bar{y}_{n+1}$ as an estimate of the local truncation error of $\bar{y}_{n+1}$.

It turns out that such $\hat{y}_{n+1}$ and $\hat{\gamma}_j$ exist, but the number of stages q' required to obtain both $\hat{y}_{n+1}$ and $\bar{y}_{n+1}$ is slightly higher than the number of stages needed to obtain only $\bar{y}_{n+1}$. This is not unexpected. For example, the classical fourth-order RK method requires only four stages while the Fehlberg formulas of order four and five need six stages, but the latter include an useful error estimator for the fourth-order value. In Gear [4], it is shown that fourth-order formulas given in (1.4) are possible with only six stages. In the following, we will show that with one more function evaluation, one can generate both fourth-order values for the scaled derivatives and a fifth-order value for $y(t_{n+1})$ from which an error estimator can be formulated as described above.

We begin by deriving fourth-order formulas using six function evaluations ($p = 4$, $q' = 6$ in (1.6a) and (1.6b), for convenience we consider f as a function of y only.) If we express the values $h_R^s y^{(s)}$, $s=1,\ldots,4$ in terms of their elementary differentials (see Gear [3],[5],)

$$\begin{aligned} h_R y' &= h_R f, \\ h_R^2 y^{(2)} &= h_R^2 f_1 f, \\ h_R^3 y^{(3)} &= h_R^3 [f_2 f^2 + f_1^2 f], \\ h_R^4 y^{(4)} &= h_R^4 [f_3 f^3 + 3f_2 f_1 f^2 + f_1 f_2 f^2 + f_1^3 f], \end{aligned}$$

and expand k_i, $i=0,\ldots,5$ in terms of elementary differentials of order up to four, for example,

$$\begin{aligned} k_0 &= h_R f, \\ k_1 &= h_R f + \alpha_1 h_R^2 f_1 f + \frac{1}{2!} \alpha_1^2 h_R^3 f_2 f^2 + \frac{1}{3!} \alpha_1^3 h_R^4 f_3^3 + O(h_R^5) \end{aligned}$$

and so on ..., then to obtain fourth-order approximations of the scaled derivatives $h_R^s y^{(s)}/s!$, $s=1,\ldots,4$, the following identity must be satisfied by β_{ij} and γ_{js},

$$A\,\Gamma = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2!} & 0 & 0 \\ 0 & 0 & \frac{2}{3!} & 0 \\ 0 & 0 & \frac{1}{3!} & 0 \\ 0 & 0 & 0 & \frac{6}{4!} \\ 0 & 0 & 0 & \frac{3}{4!} \\ 0 & 0 & 0 & \frac{2}{4!} \\ 0 & 0 & 0 & \frac{1}{4!} \end{bmatrix} \qquad (2.1)$$

where Γ is the 6×4 matrix $[\gamma_{js}]$ and A is an 8×6 mat ix whose rows correspond to the terms f, f_1f, f_2f^2, f_1^2f, f_3f^3, $f_2f_1f^2$, $f_1f_2f^2$ and f_1^3f respectively, and whose columns correspond to k_i, $i=0,\ldots,5$. A is found to be

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 \\ 0 & \alpha_1^2 & \alpha_2^2 & \alpha_3^2 & \alpha_4^2 & \alpha_5^2 \\ 0 & 0 & P_2 & P_3 & P_4 & P_5 \\ 0 & \alpha_1^3 & \alpha_2^3 & \alpha_3^3 & \alpha_4^3 & \alpha_5^3 \\ 0 & 0 & \alpha_2P_2 & \alpha_3P_3 & \alpha_4P_4 & \alpha_5P_5 \\ 0 & 0 & Q_2 & Q_3 & Q_4 & Q_5 \\ 0 & 0 & 0 & R_3 & R_4 & R_5 \end{bmatrix}$$

with P_i, Q_i, R_i lefined as

$$\begin{aligned} P_i &= \sum_{j=2}^{i} \beta_{ij}\alpha_{j-1}, & i &= 2,3,4,5, \\ Q_i &= \sum_{j=2}^{i} \beta_{ij}\alpha_{j-1}^2, & i &= 2,3,4,5, \\ R_i &= \sum_{j=3}^{i} \beta_{ij}P_{j-1}, & i &= 3,4,5. \end{aligned} \tag{2.2}$$

It is clear from (2.1) that the first column of Γ is equal to $[1,0,0,0,0,0]^T$ and for $s = 2,3.4$, $\gamma_{1s} = -\sum_{j=2}^{6}\gamma_{js}$. So, after eliminating the first row and column in A and the right-hand side of (2.1), there are only 21 conditions to be satisfied by 30 unknowns β_{ij}, $i=1,\ldots,5$, $j=1,\ldots,i$, and γ_{js}, $j=2,\ldots,6$, $s=2,3,4$. A solution of (2.1) can be found. In fact, Gear [4] showed the existence of a nine-parameter family of solutions.

Now, in addition to the fourth-order approximations of the scaled derivatives, is it possible to estimate the local truncation error of

$$\bar{y}_{n+1} = y_n + \sum_{s=1}^{4}\sum_{j=1}^{6} k_{j-1}\gamma_{js} \tag{2.3}$$

still using the same six stages k_i? In other words, if we denote the principal truncation error term of (2.3) by $h_R^5\phi(t_n,y(t_n))$, does there exist a non-zero vector $\hat{\gamma} = [\hat{\gamma}_1,\ldots,\hat{\gamma}_6]^T$ such that

$$\sum_{j=1}^{6} k_{j-1}\hat{\gamma}_j = h_R^5\phi(t_n,y(t_n)) + O(h_R^6)? \tag{2.4}$$

The answer is negative and a proof by contradiction is given below.

Assume that we can find A and Γ satisfying (2.1), and assume further that (2.4) holds for some non-zero vector $\hat{\gamma}$. Since the columns in the right-hand side of (2.1) are linearly independent, it follows that the columns of Γ, denoted as $\gamma^{(s)}$ for $s=1,\ldots,4$, are linearly independent. Also, since (2.4) implies that $A\hat{\gamma} = 0$, $\hat{\gamma}$ must be linearly independent of $\gamma^{(s)}$. Otherwise, there exist scalars ξ_s, not all of them zero, such that

$$\sum_{s=1}^{4} \xi_s A\gamma^{(s)} = A\hat{\gamma} = 0,$$

which violates the linear independency of the columns in the right-hand side of (2.1).

Let B be the 4 × 6 matrix consisting of the last four rows of A. Then the four independent vectors $\gamma^{(1)}$, $\gamma^{(2)}$, $\gamma^{(3)}$ and $\hat{\gamma}$ belong to the null space of B. Therefore, B has rank at most 2. If $\alpha_1 = 0$, the method reduces to a five-stage method which is non-existent (see Gear [4],) so we can assume that $\alpha_1 \neq 0$. Thus the first row of B is independent of the remaining rows because of the zeros in A_{12}, $i=6,7,8$. Hence these rows assume rank at most 1. Since $A_{83} = 0$, this is possible only if $\alpha_2 P_2 = Q_2 = 0$. From (2.2), we have $Q_2 = P_2\alpha_1$, so $P_2 = 0$ since $\alpha_1 \neq 0$. Again from (2.2), this implies that $R_3 = \beta_{33}P_2 = 0$, which then leads to $\alpha_3 P_3 = Q_3 = 0$ as a consequence of the linear dependency of the last three rows of B. For $\alpha_3 P_3$ to be 0, either $\alpha_3 = 0$ or $P_3 = 0$, but not both. Otherwise, a five-stage method results. So assume that $\alpha_3 = 0$ and consider the following subsystem of (2.1)

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_4 & \alpha_5 \\ \alpha_1^2 & \alpha_2^2 & \alpha_4^2 & \alpha_5^2 \\ \alpha_1^3 & \alpha_2^3 & \alpha_4^3 & \alpha_5^3 \\ 0 & 0 & \alpha_4 P_4 & \alpha_5 P_5 \end{bmatrix} \overline{\Gamma} = \begin{bmatrix} \frac{1}{2!} & 0 & 0 \\ 0 & \frac{2}{3!} & 0 \\ 0 & 0 & \frac{6}{4!} \\ 0 & 0 & \frac{3}{4!} \end{bmatrix} \tag{2.5}$$

where $\overline{\Gamma}$ is Γ after crossing out the first column and the first and fourth rows. It is fairly clear that the matrix in the left-hand side of (2.5) i non-singular. Since $A\gamma = 0$, this non-singularity implies that $\gamma_i = 0$, i=2,3,5,6. But then, the first and fourth rows of A imply that γ is identically zero, which is a contradiction. Hence we must have $P_3 = 0$. As a result, $R_4 = \beta_{43}P_2 + \beta_{44}P_3 = 0$, and consequently, $\alpha_4 P_4 = Q_4 = 0$. Now, P_4 cannot be 0. Otherwise, R_5 is 0 and hence (2.1) cannot be satisfied. So, $\alpha_4 = 0$. Again, consideration of a subsystem of (2.1) similar to (2.5) leads to the conclusion that γ is zero. Our proof is thus complete.

Since using only six function evaluations is not enough to obtain the desired error estimator, we have to do some extra function evaluations, but how many? It turns out that we need only one more. When the coefficients β_{ij} given by the Fehlberg formulas of order four and five are substituted in the matrix A, we discover that it is not consistent with the right-hand side of (2.1), thus cannot be solved for Γ. However, by changing the parameter R_5 in the sixth column of A, the inconsistency can be removed. Consequently, if we still maintain a six-stage RK process, but evaluate the sixth stage twice: the first time to obtain k_5 and use it with k_i, $i=0,\dots,4$ to solve for Γ; the second time to get k_5^* which, together with k_i, $i=0,\dots,4$, gives rise to a vector $\tilde{\gamma} = [\tilde{\gamma}_1,\dots,\tilde{\gamma}_6]^T$ so that the following

$$\tilde{y}_{n+1} = y_n + \sum_{j=1}^{5} k_{j-1}\tilde{\gamma}_j + k_5^*\tilde{\gamma}_6,$$

$$\tilde{y}_{n+1} = y(t_{n+1}) + O(h_R^6),$$

holds, then our goal is achieved.

Using this approach, a set of coefficients β_{ij} and γ_{js} has been calculated and is given in Table 2.1 and Table 2.2. And the error estimator to be used is defined by

$$\sum_{j=1}^{5} k_{j-1}\tilde{\gamma}_j + k_5^*\tilde{\gamma}_6 - \sum_{s=1}^{4}\sum_{j=1}^{6} k_{j-1}\gamma_{js}.$$

What remains to be discussed now is how to develop a step adjustment scheme which, based on the information provided by the stepsize h_R and the error estimator, calculates the stepsize h that is actually taken. Also needed is a strategy to select h_R for the next step. At this time, our knowledge on the subject relies more on experimentation and heuristics than on conclusive mathematical analysis. So, we postpone any discussion until section 4 where numerical implementation and testing are described.

Table 2.1 Coefficients β_{ij} and β^*_{5j}

β_{1j}	:	$\frac{1}{4}$				
β_{2j}	:	$\frac{3}{32}$	$\frac{9}{32}$			
β_{3j}	:	$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$		
β_{4j}	:	$\frac{439}{216}$	$-\frac{1}{8}$	$\frac{3680}{513}$	$-\frac{845}{4104}$	
β_{5j}	:	$\frac{289}{216}$	$-\frac{14}{3}$	$\frac{2072}{513}$	$\frac{169}{1026}$	$-\frac{3}{8}$
β^*_{5j}	:	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$

Table 2.2 Coefficients $\gamma^{(s)}$ and $\tilde{\gamma}$

$\gamma^{(1)}$	$\gamma^{(2)}$	$\gamma^{(3)}$	$\gamma^{(4)}$	$\tilde{\gamma}$
1	$-\frac{201}{80}$	$\frac{1393}{540}$	$-\frac{137}{144}$	$\frac{16}{135}$
0	0	0	0	0
0	$\frac{7232}{1425}$	$-\frac{114688}{12825}$	$\frac{3776}{855}$	$\frac{6656}{12825}$
0	$-\frac{15379}{4560}$	$\frac{81289}{10260}$	$-\frac{10985}{2736}$	$\frac{28561}{56430}$
0	$\frac{261}{100}$	$-\frac{159}{25}$	$\frac{71}{20}$	$-\frac{9}{50}$
0	$-\frac{9}{5}$	$\frac{24}{5}$	-3	$\frac{2}{55}$

3. REGIONS OF ABSOLUTE STABILITY

When a modified six-stage fourth-order Runge-Kutta method

$$y_{n+1} = y_n + \sum_{s=1}^{4} r^s \sum_{j=1}^{6} k_{j-1}\gamma_{js} \tag{3.1}$$

is applied to the test equation $y' = \lambda y$, the value y_{n+1} at step t_{n+1} is related to y_n of the previous step by the simple expression

$$y_{n+1} = P(\mu,r)y_n$$

where $P(\mu,r)$, considered as a function of $\mu = h\lambda$, is a sixth-degree polynomial whose coefficients are functions of $r = h/h_R$. Therefore, by keeping r fixed, we can define the region of absolute stability of (3.1) in the same way as defining the region of absolute stability for a conventional RK method. It is for a fixed r, the set of all complex values μ for which $| P(\mu,r) | < 1$. In this section, we are interested in plotting these regions of absolute stability for different values of r.

To obtain $P(\mu,r)$, we first make the substitution $f(t,y) = \lambda y$ in k_i. For example,

$$\begin{aligned} k_0 &= h_R f(t_n,y_n) = h_R\lambda y_n, \\ k_1 &= h_R f(t_n + h_R\alpha_1 , y_n + \beta_{11}k_0) \\ &= h_R\lambda(y_n + \beta_{11}h_R\lambda y_n) \\ &= [h_R\lambda + \alpha_1(h_R\lambda)^2]y_n. \end{aligned}$$

In general, we have

$$k_i = \sum_{m=1}^{i+1} \delta_{im}(h_R\lambda)^m \, y_n, \quad i = 0,\ldots,5 \tag{3.2}$$

where δ_{im}, i=C,...,5, $m=1,\ldots,i+1$ are found to be

$$\begin{array}{llllllll}
\delta_{0m} & : & 1 & & & & & \\
\delta_{1m} & : & 1 & \alpha_1 & & & & \\
\delta_{2m} & : & 1 & \alpha_2 & P_2 & & & \\
\delta_{3m} & : & 1 & \alpha_3 & P_3 & R_3 & & \\
\delta_{4m} & : & 1 & \alpha_4 & P_4 & R_4 & V_4 & \\
\delta_{5m} & : & 1 & \alpha_5 & P_5 & R_5 & V_5 & W_5
\end{array}$$

with α_i, P_i, R_i defined as in (1.2) and (2.2), and

$$V_i = \sum_{j=4}^{i} \beta_{ij}R_{j-1}, \quad i = 4,5,$$

$$W_5 = \beta_{55}V_4.$$

Now, using (3.2) to replace k_i in (3.1), we get

$$\begin{aligned}
y_{n+1} &= y_n + \sum_{s=1}^{4} r^s \sum_{j=1}^{6} \Big[\sum_{m=1}^{j} \delta_{j-1,m}(h_R\lambda)^m \, y_n\Big]\gamma_{js} \\
&= \Big[1 + \sum_{s=1}^{4} r^s \sum_{m=1}^{6} \sum_{j=m}^{6} \delta_{j-1,m}\gamma_{js}(h_R\lambda)^m\Big]y_n \\
&= P(\mu,r)y_n.
\end{aligned}$$

Since γ_{js} satisfy (2.1) and in particular, $\gamma_{j1} = [1,0,0,0,0,0]^T$, it follows that for $s = 1$,

$$r \sum_{m=1}^{6} \sum_{j=m}^{6} \delta_{j-1,m}\gamma_{j1}(h_R\lambda)^m = r(h_R\lambda) = h\lambda = \mu,$$

and for $s = 2,3,4$,

$$r^s \sum_{m=1}^{6} \sum_{j=m}^{6} \delta_{j-1,m}\gamma_{js}(h_R\lambda)^m = r^s[\frac{1}{s!}(h_R\lambda)^s + (\gamma_{5s}V_4 + \gamma_{6s}V_5)(h_R\lambda)^5 + \gamma_{6s}W_5(h_R\lambda)^6] = \frac{1}{s!}\mu^s + (\gamma_{5s}V_4 + \gamma_{6s}V_5)r^{s-5}\mu^5 + \gamma_{6s}W_5 r^{s-6}\mu^6.$$

Consequently,

$$P(\mu,r) = 1 + \mu + \frac{1}{2!}\mu^2 + \frac{1}{3!}\mu^3 + \frac{1}{4!}\mu^4 + [\sum_{s=2}^{4}(\gamma_{5s}V_4 + \gamma_{6s}V_5)r^{s-5}]\mu^5 + [\sum_{s=2}^{4}\gamma_{6s}W_5 r^{s-6}]\mu^6.$$

To plot the region of stability, we set

$$P(\mu,r) = e^{i\theta}, \quad \theta \in [0,2\pi] \tag{3.3}$$

and solve for μ. This gives the boundary of the region of stability since all those μ for which (3.3) holds, are such that $| P(\mu,r) | = 1$. The boundary divides the μ-complex plane into regions and it is not difficult to determine which one is a region of stability. One can solve (3.3) most conveniently by finding the zeros of the polynomial

$$Q(\mu,\theta) = P(\mu,r) - e^{i\theta}, \quad \theta \in [0,2\pi],$$

using the Newton method for example. We first divide the interval $[0,2\pi]$ into N subintervals, and then find the roots μ_n of $Q(\mu,\theta_n) = 0$, $\theta_n = nh$, $h = 2\pi/N$ by taking an initial guess

$$\mu_{n,(0)} = \mu_{n-1},$$

and iterating as follows,

$$\mu_{n,(m+1)} = \mu_{n,(m)} - \frac{Q(\mu_{n,(m)},\theta_n)}{Q'(\mu_{n,(m)},\theta_n)}.$$

Since Q is a polynomial of degree six, there are six branches of the boundary to be traced, each of which starts at a root of $Q(\mu,0) = 0$. We know that for any r, $Q(0,0) = P(0,r) - 1 = 0$. So, we can always start at the origin of the μ-complex plane and plot the boundary until it forms a closed curve. However, in some cases, the region of stability is disconnected. We then must find another starting point and proceed as above.

Another useful fact about the region of stability is that it is symmetric about the real axis. This is true because the coefficients of P are real. So, if μ satisfies (3.3), that is $P(\mu,r) = e^{i\theta}$, then $P(\bar{\mu},r) = \bar{P}(\mu,r) = e^{-i\theta} = e^{i(2\pi-\theta)}$, thus $\bar{\mu}$ also satisfies (3.3). Therefore, we need only plot the boundary in the upper half-plane.

In Figures 3.1 through 3.10, regions of stability of (3.1) are drawn for various values of r ranging from 0.1 to 10000. The coefficients used are taken from Tables 2.1 and 2.2. Of course, a practical range of r is more restricted. Still, it is interesting to see how one can enlarge or shrink a region of stability just by changing r. The following facts are observed:

1. In the range of r mentioned above, there are intervals in which the region of stability either expands or shrinks. Also, there are values of r where a whole region is split into separate pieces, and values of r where disconnected subregions are joined back into one. To best illustrate this, various regions of stability in each interval are selected and put in different figures.

2. For $r = 0.525$, 0.9741 and 43, the region of stability has a large intersect with the real axis, as can be seen in Figures 3.2, 3.4 and 3.8. For $r < 0.515$ and $1 < r < 18$, the region of stability consists of three disjoint regions, a large one contained in the negative half-plane and two smaller ones located in the positive half-plane. This is shown in Figures 3.1, 3.6 and 3.7. Note tnat the region for $r = 0.1$ is extremely small, compared to that of a conventional method. For values of r near 2, it seems that the regions are approaching a limiting region, but then after $r = 2$, the same patt n of expanding and shrinking recurs.

3. As $r \to \infty$, the region of stability of (3.1) becomes more and more identical with that of a conventional four-stage fourth-order RK method. One can easily see this by looking at the expression of $P(\mu,r)$: as r becomes large, the coefficients of μ^5 and μ^6 both approach 0. In Figure 3.10, the regions for $r = 50, 100, 1000$ and 10000 are plotted. For larger values, it is almost impossible for the eyes to distinguish between two different regions.

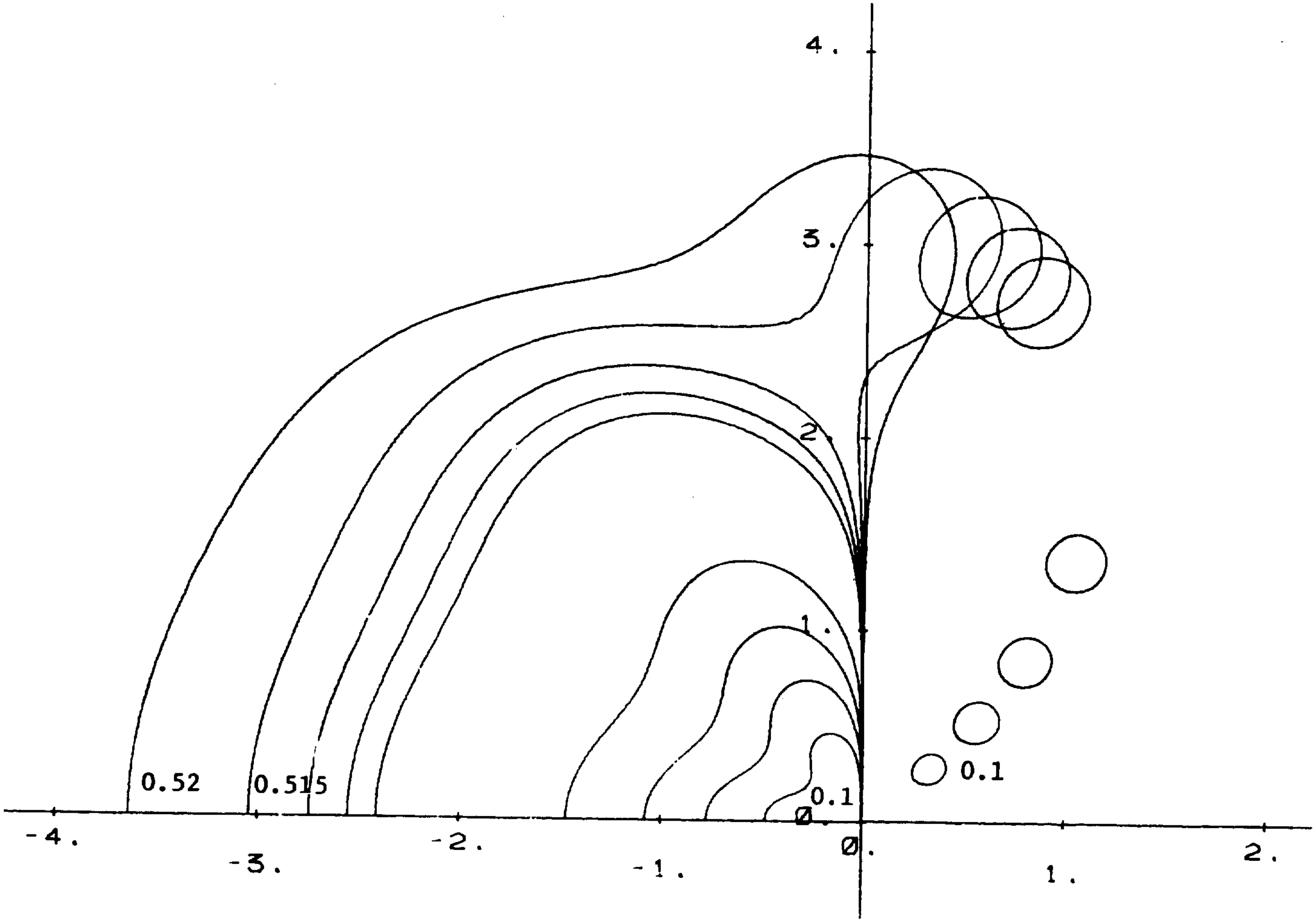

Figure 3.1 Regions of stability for r ranging from 0.1 to 0.52

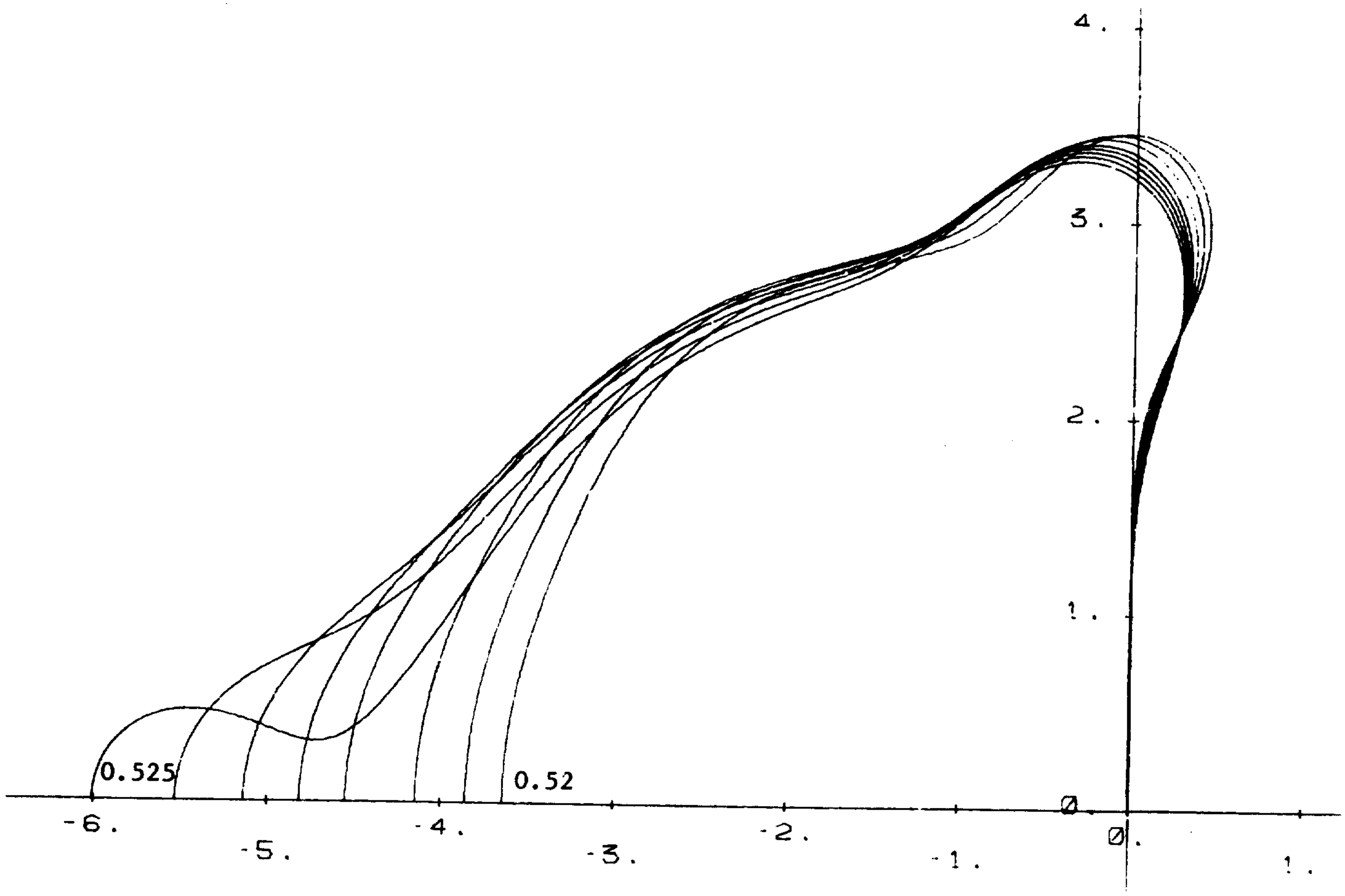

Figure 3.2 Regions of stability for r ranging from 0.52 to 0.525

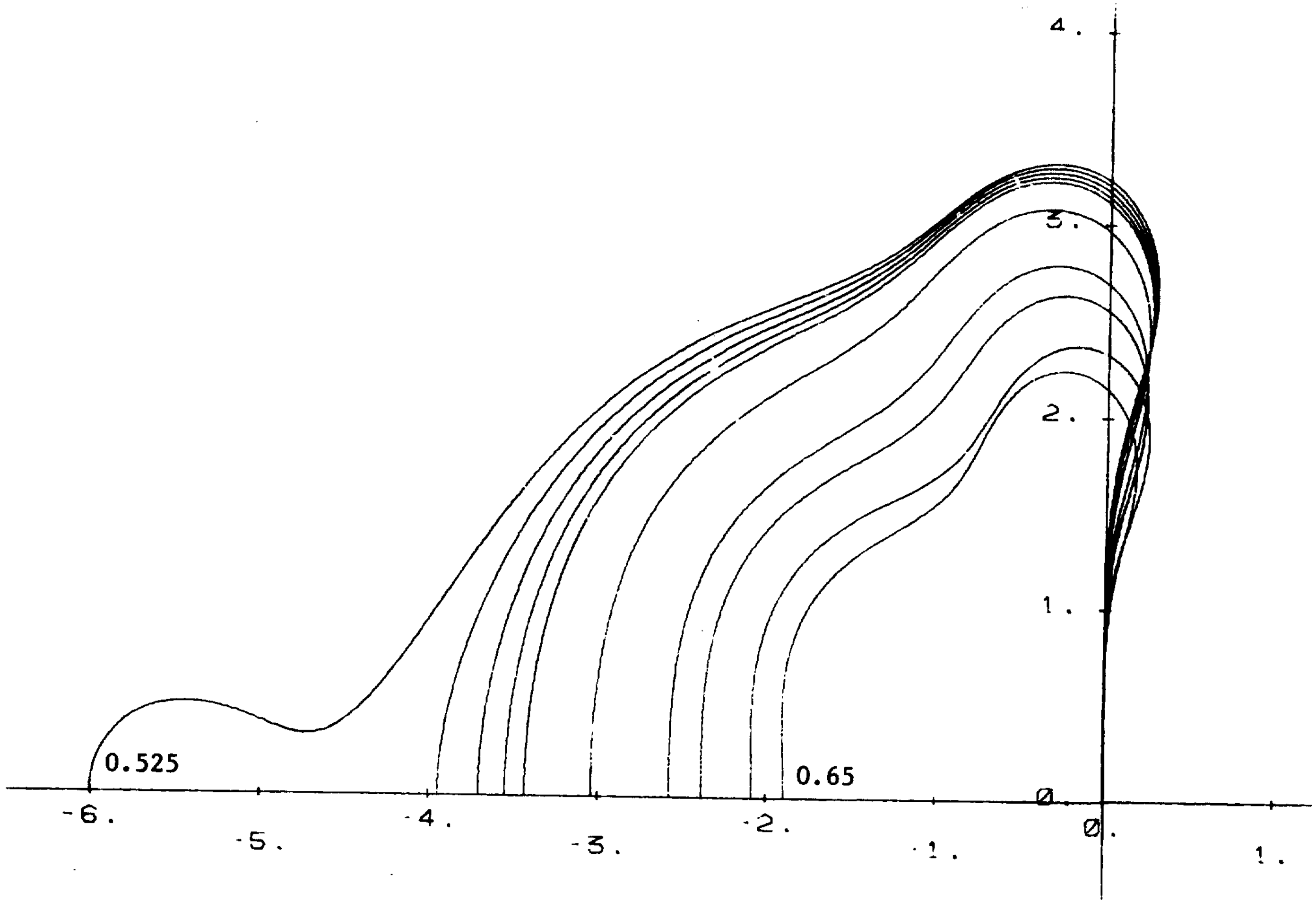

Figure 3.3 Regions of stability for r ranging from 0.525 to 0.65

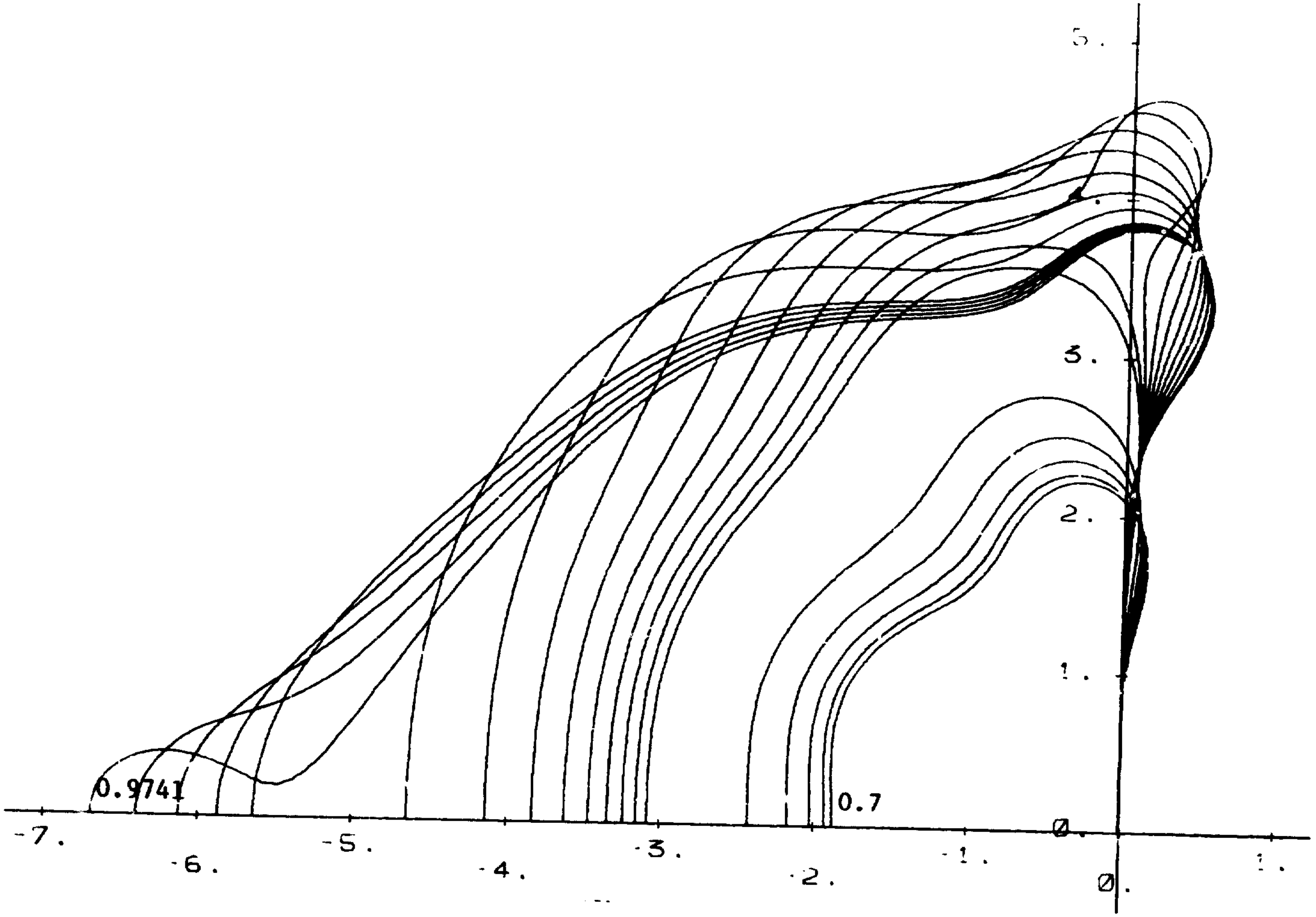

Figure 3.4 Regions of stability for r ranging from 0.7 to 0.9741

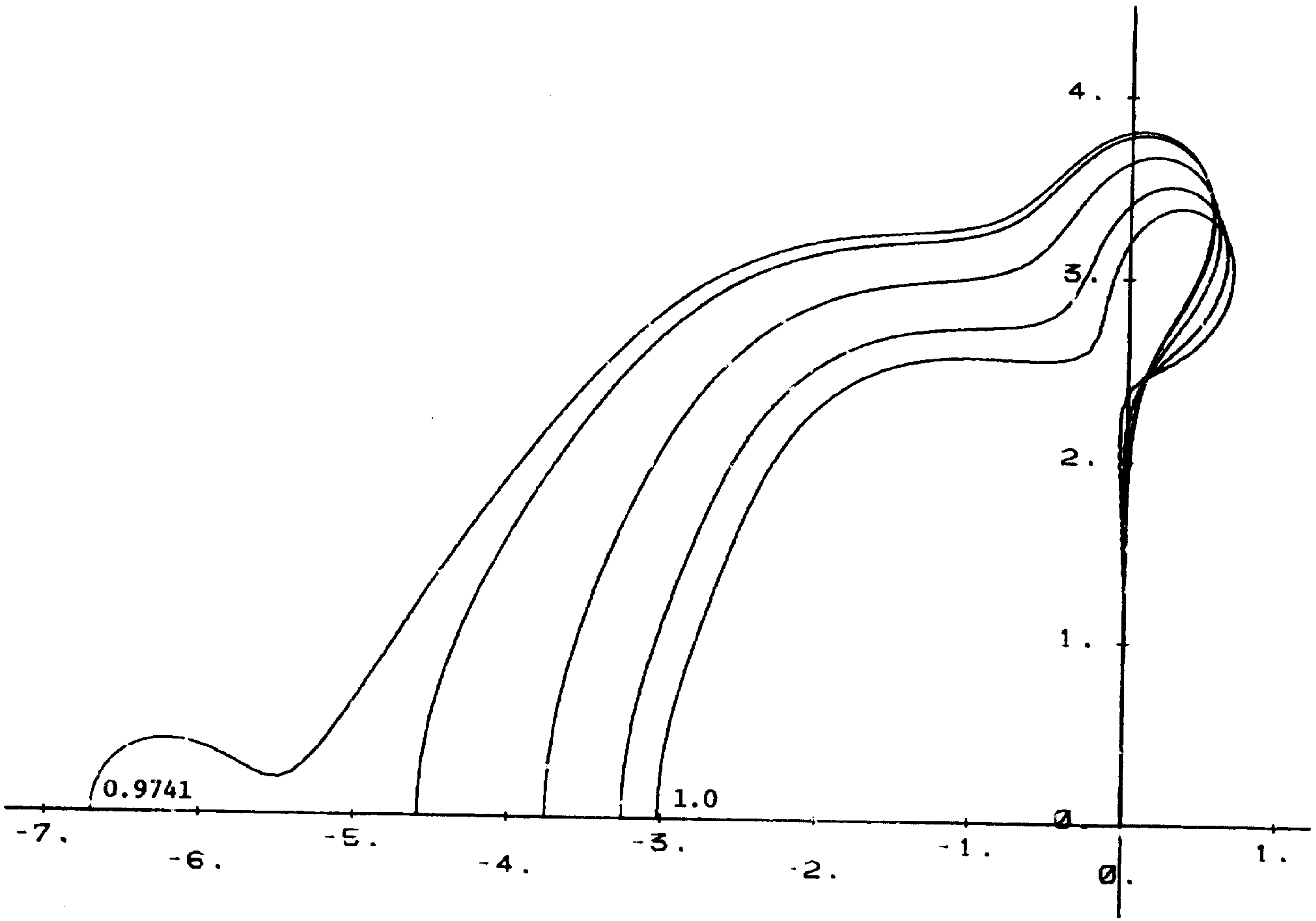

Figure 3.5 Regions of stability for r ranging from 0.9741 to 1.0

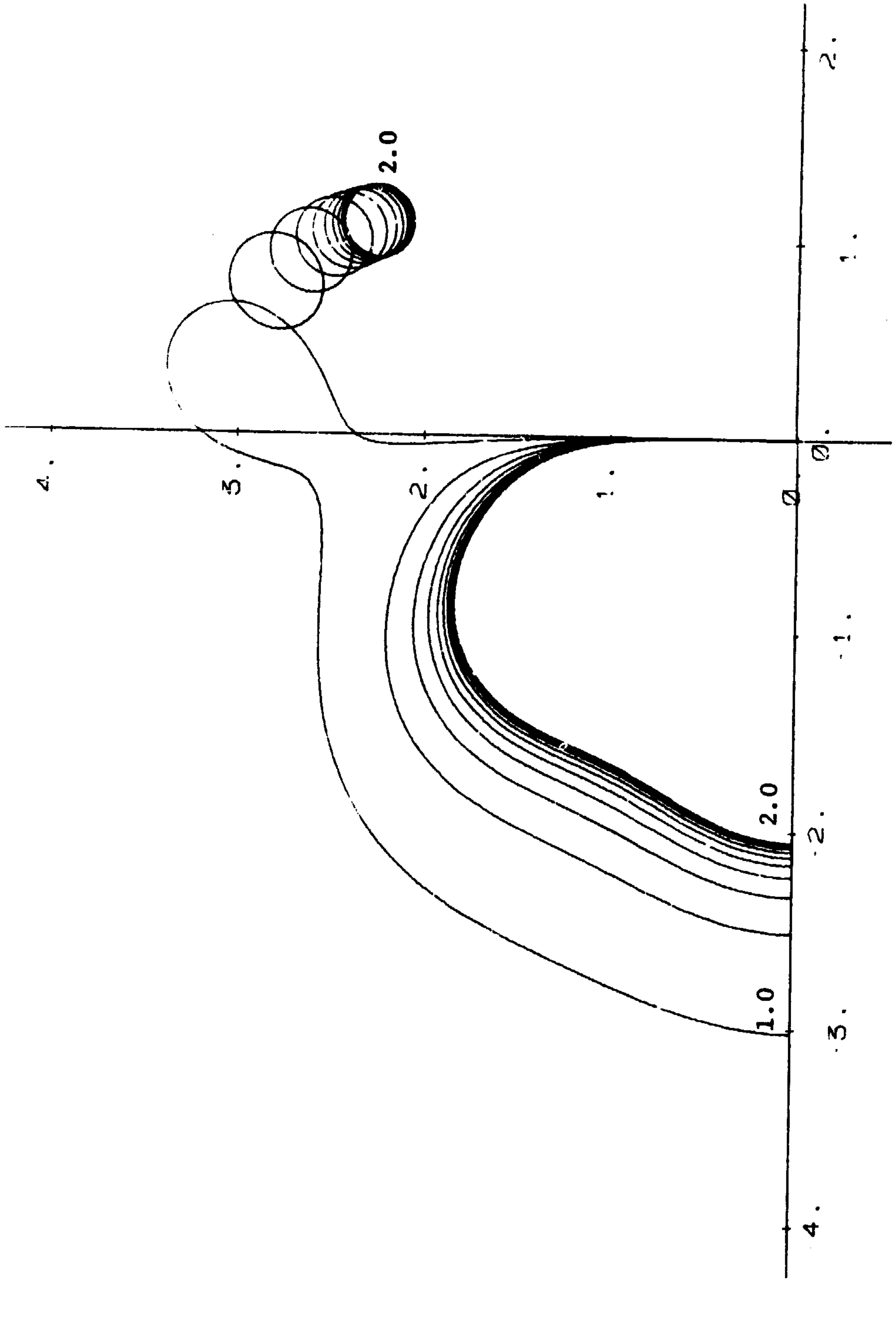

Figure 3.6 Regicns of stability for r ranging from 1.0 to 2.0

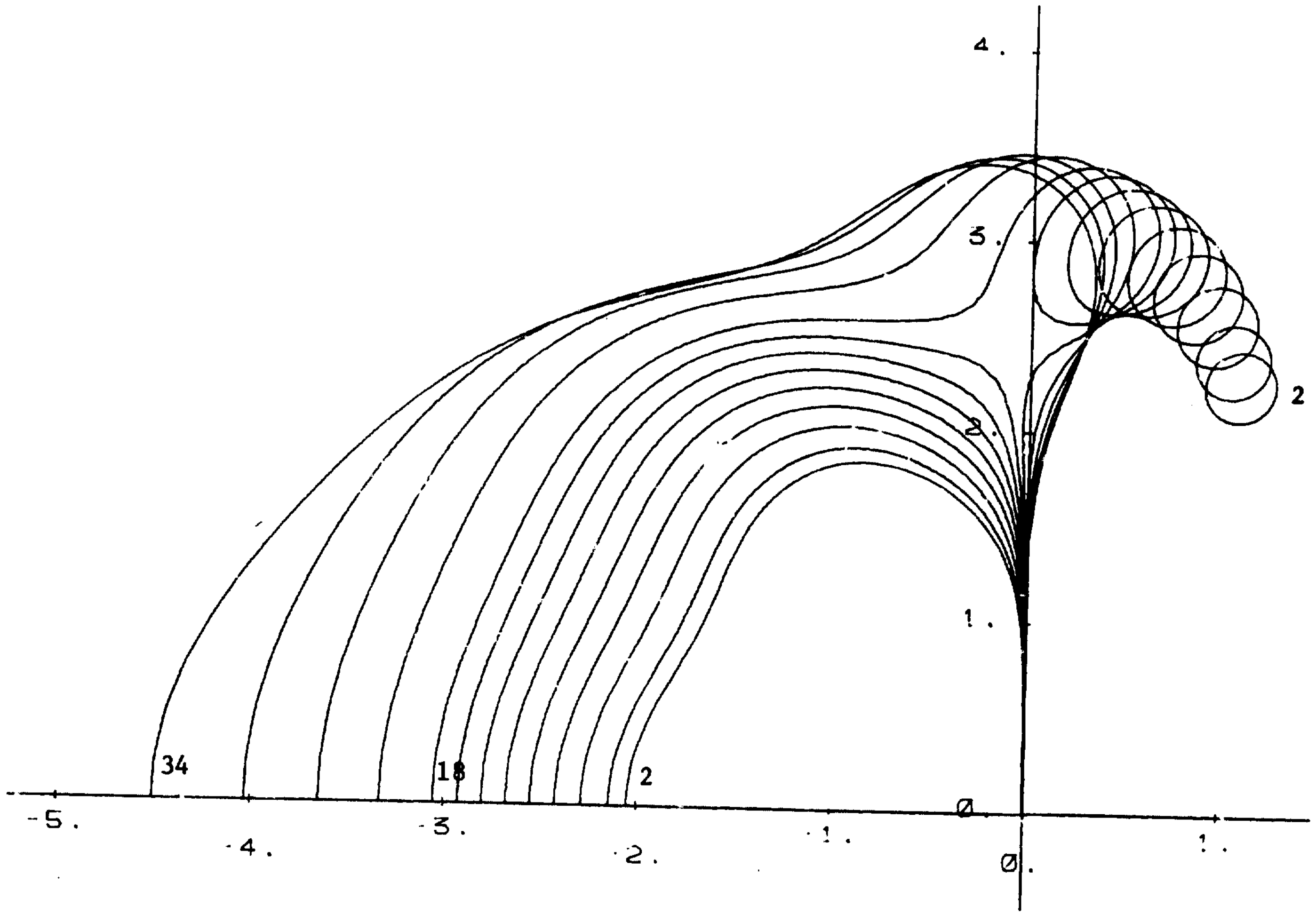

Figure 3.7 Regions of stability for r ranging from 2 to 34

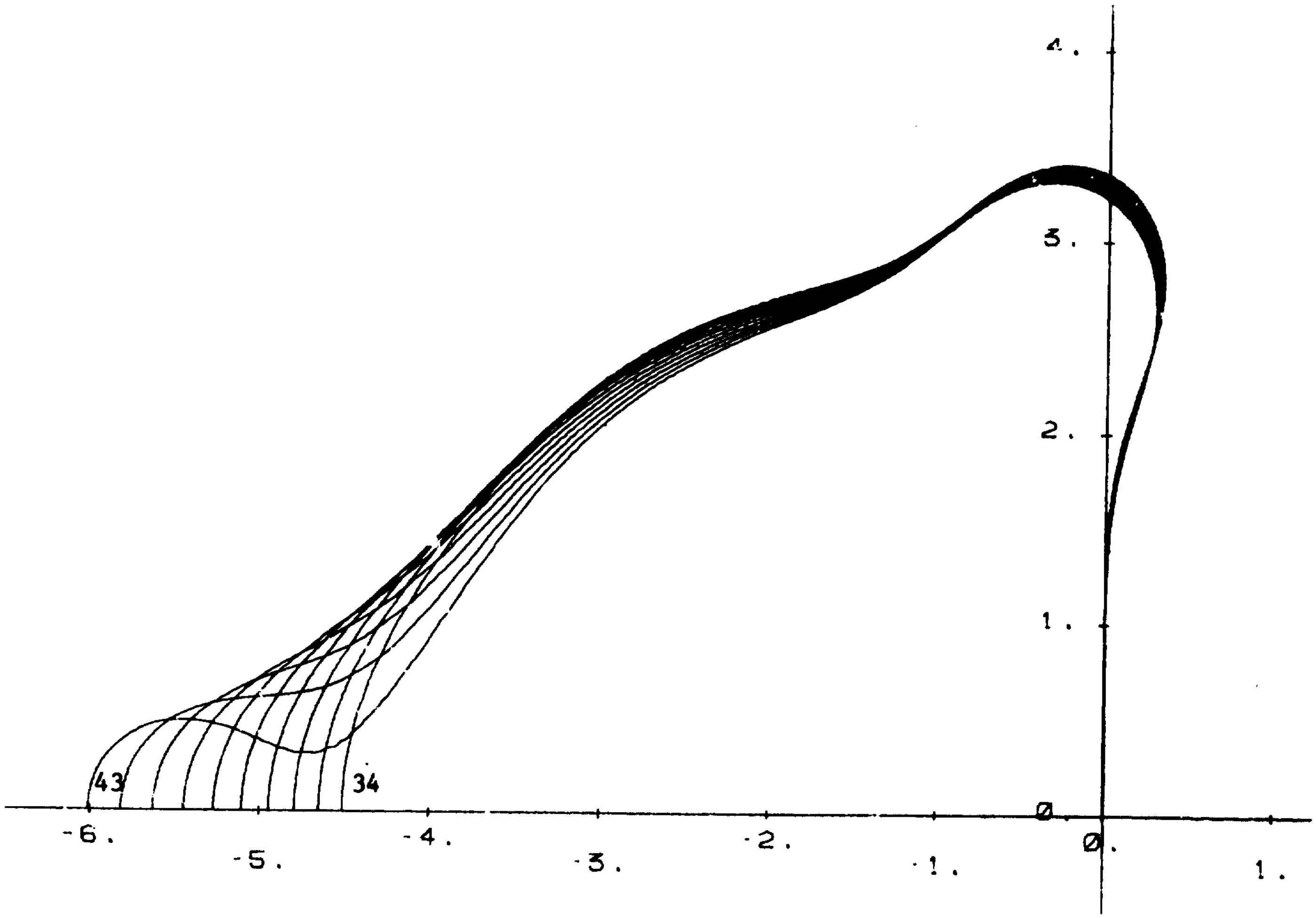

Figure 3.8 Regions of stability for r ranging from 34 to 43

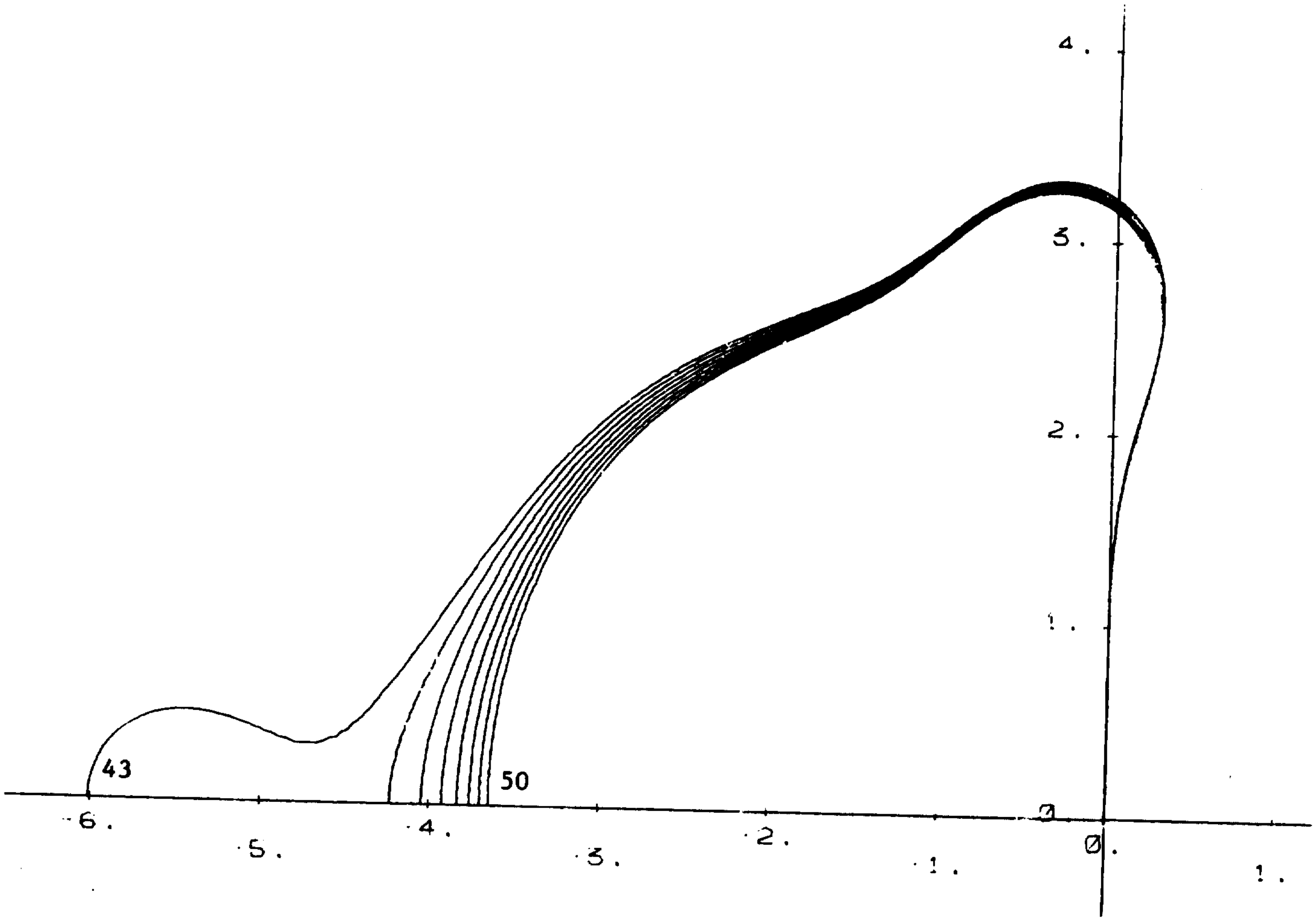

Figure 3.9 Regions of stability for r ranging from 43 to 50

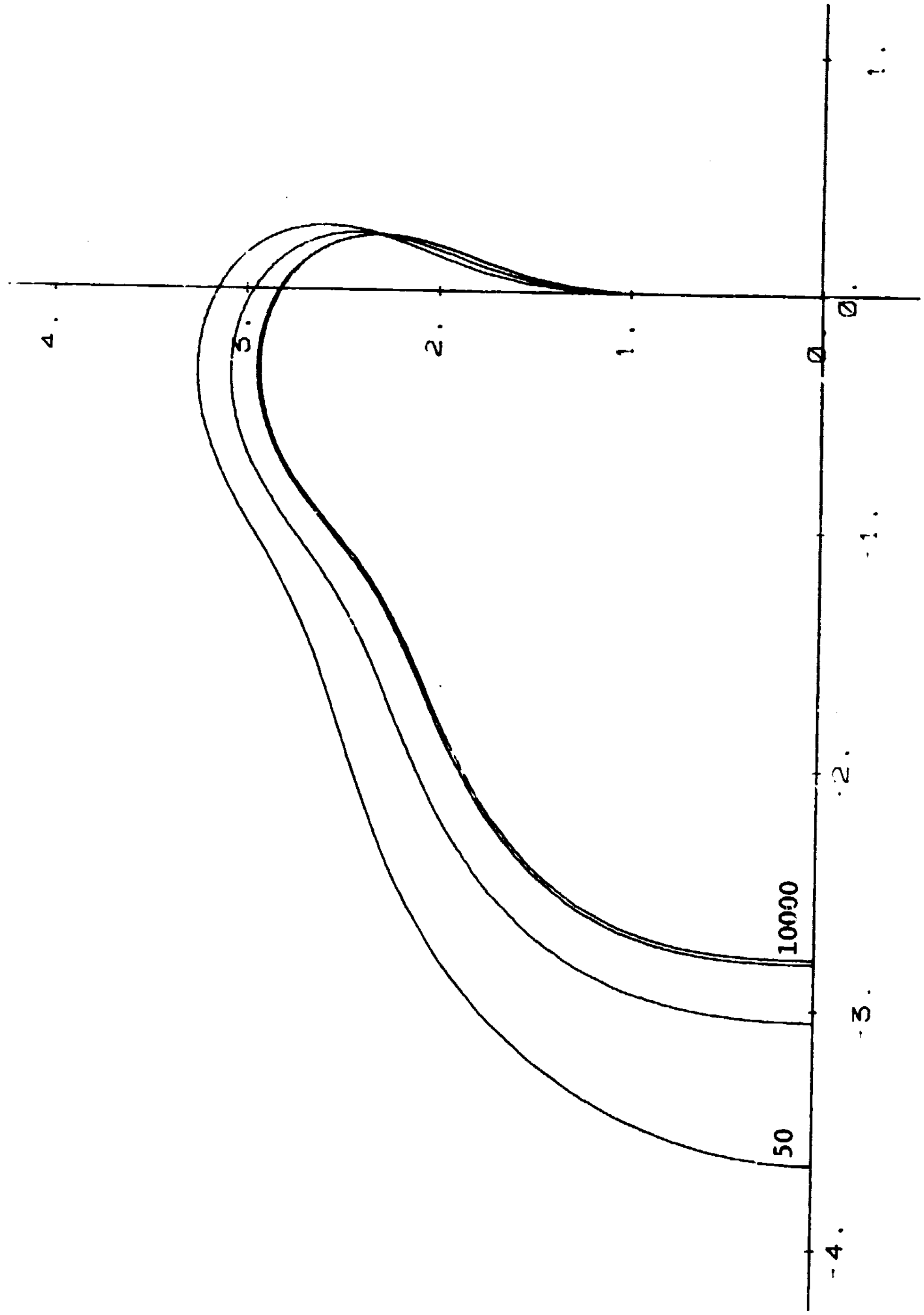

Figure 3.10 Regions of stability for r ranging from 50 to 10000

4. NUMERICAL IMPLEMENTATION

In addition to the error estimator obtained in section 2, any useful implementation of a modified six-stage fourth-order Runge-Kutta method must also include algorithms to adjust the stepsize h and to select h_R for the next step. Due to the limited information we know about the differential equation, we are unable to offer a complete analysis which guarantees the best selection of h or h_R. We can discuss instead some practical techniques which choose h in an attempt to control the local truncation error.

We recall that the local truncation error committed by taking a stepsize h_R can be written as

$$h_R^5\phi(t_n,y(t_n)) + O(h_R^6) \tag{4.1}$$

where $\phi(t,y)$ depends on the method coefficients and the elementary differentials of order five evaluated at $(t_n,y(t_n))$. Now, suppose that the local truncation error of

$$y_{n+1} = \sum_{s=1}^{4} r^s \sum_{j=1}^{6} k_{j-1}\gamma_{js} \tag{4.2}$$

computed with the adjusted stepsize h, can be expressed similarly as

$$h^5\phi(t_n,y(t_n)) + O(h^6) \tag{4.3}$$

and suppose that the first term in both (4.1) and (4.3) is dominant. Then since we have at our disposition an estimate of (4.1), we simply take h so that

$$r^5 || \text{ error estimate } || < \varepsilon \qquad (4.4)$$

for some given error tolerance ε. Or, equivalently,

$$h = h_R[\frac{\varepsilon}{|| \text{ error estimate } ||}]^{1/5}\omega$$

where ω is a "safety" factor less than 1. Though it is simple, (4.3) does not hold. The best we can hope for is that it is close to the true local truncation error. However, once h is taken, there is no available test for us to know whether y_{n+1} is acceptable.

Instead of (4.3), we have the following local truncation error

$$h^5\phi(t_n,y(t_n),r) + 0(h^6) \qquad (4.5)$$

where $\phi(t,y,r)$ depends not only on the coefficients and the differentials, but also on $r = h/h_R$. Thus, if we want to use (4.4), we must also choose r subject to the following constraint

$$||\phi(t_n,y(t_n),r)|| < ||\phi(t_n,y(t_n))||. \qquad (4.6)$$

The problem posed by (4.6) is hard to solve, if not impossible, since the elementary differentials are generally not known. However, in the case of linear initial value problems, an answer to (4.6) is as follows.

Let $E_{5,j}$, $j=1,\ldots,9$ denote the fifth-order elementary differentials f_4f^4, $f_3f_1f^3$, $f_1f_3f^3$, $f_2^2f^3$, $f_2f_1^2f^2$, $f_2(f_1f)^2$, $f_1f_2f_1f^2$, $f_1^2f_2f^2$ and f_1^4f

respectively. If we include these fifth-order terms in the Taylor series expansion of k_i, $i=0,\ldots,5$ and let C be the 9 × 6 matrix whose rows correspond to $E_{5,j}$, $j=1,\ldots,9$ and whose columns correspond to k_i, $i=0,\ldots,5$,

$$C = \begin{bmatrix} 0 & \alpha_1^4 & \alpha_2^4 & \alpha_3^4 & \alpha_4^4 & \alpha_5^4 \\ 0 & 0 & \alpha_2^2 P_2 & \alpha_3^2 P_3 & \alpha_4^2 P_4 & \alpha_5^2 P_5 \\ 0 & 0 & S_2 & S_3 & S_4 & S_5 \\ 0 & 0 & \alpha_2 Q_2 & \alpha_3 Q_3 & \alpha_4 Q_4 & \alpha_5 Q_5 \\ 0 & 0 & 0 & \alpha_3 R_3 & \alpha_4 R_4 & \alpha_5 R_5 \\ 0 & 0 & P_2^2 & P_3^2 & P_4^2 & P_5^2 \\ 0 & 0 & 0 & T_3 & T_4 & T_5 \\ 0 & 0 & 0 & U_3 & U_4 & U_5 \\ 0 & 0 & 0 & 0 & V_4 & V_5 \end{bmatrix}$$

with α_i, P_i, Q_i, R_i and V_i defined as in the previous sections, and

$$S_i = \sum_{j=2}^{i} \beta_{ij}\alpha_{j-1}^3, \qquad i = 2,3,4,5,$$

$$T_i = \sum_{j=3}^{i} \beta_{ij}\alpha_{j-1}P_{j-1}, \qquad i = 3,4,5,$$

$$U_i = \sum_{j=3}^{i} \beta_{ij}Q_{j-1}, \qquad i = 3,4,5.$$

then we can explicitly express ψ and ϕ as

$$\psi(t_n,y(t_n)) = \sum_{j=1}^{9} [\tilde{\tau}_1 - \sum_{s=2}^{4} \tau_{sj}]E_{5,j}$$

and

$$\phi(t_n, y(t_n)) = \sum_{j=1}^{9} [\tilde{\tau}_1 - \sum_{s=2}^{4} r^{s-5} \tau_{sj}] E_{5,j}$$

where $\tilde{\tau}_j$ and τ_{sj} are the components of the vectors $C\tilde{\gamma}$ and $C\gamma^{(s)}$, $s=2,\ldots,4$, respectively. Since φ and ϕ are combinations of $E_{5,j}$, it is not easy to determine r so that (4.6) is satisfied. But, when f(t,y) is linear in y, the $E_{5,j}$ terms are all zero, except $E_{5,9} = f_1^4 f$. Thus, the constraint (4.6) is simplified to

$$|\Phi(\rho)| < |\Phi(1)| \qquad (4.7)$$

where $\Phi(\rho) = \tilde{\tau}_9 - \tau_{49}\rho - \tau_{39}\rho^2 - \tau_{29}\rho^3$, a cubic polynomial in $\rho = \frac{1}{r}$. The set of coefficients discovered in section 2 unfortunately yields a small range of solution in the neighborhood of r = 1. However, it is believed that coefficients can be found to enlarge this range.

Thus, for linear problems, we choose r to satisfy (4.4) and (4.7). For other problems, since (4.4) is the only piece of information we possess, we have no other choice but to calculate r from (4.4) with a rather small "safety" factor ω. As for the selection of h_R for the next step, we can either compute a weighted average of the previous h_R and h, or take h_R as a scaled value of h. It should be mentioned that these strategies are dictated by experimentation and appear to be better than other alternatives that we have tried.

We have implemented the modified six-stage fourth-order RK method and its related techniques into an experimental code and done some tests. The set of problems we solved consists of the linear problem

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

integrated over the interval [0,1], and the non-linear problem

$$y' = (y - \sin(t)) - (y - \sin(t))^2 + \cos(t), \quad y(0) = 0.5,$$

integrated over the interval [0,10]. We also solved these two problems using the code RKF45, an implementation of the Fehlberg fourth-fifth order Runge-Kutta method. The performances of the two codes are compared and presented in Table 4.1 and Table 4.2 for the linear and non-linear problems, respectively. The following statistics are shown for each method:

EPS	error tolerance
NSTEP	total number of steps taken
NFCN	total number of function evaluations
RELERR	maximum relative error for the linear problem
ABSERR	maximum absolute error for the non-linear problem

To maintain a certain degree of fairness in comparing the two methods, we used the same initial stepsizes and the same norm as used by RKF45.

In both problems, the modified RK method produced slightly less accurate results for high-order tolerances. This was probably caused by the failure of the code to adjust the stepsize efficiently although it

Table 4.1 Numerical results for the linear problem

EPS	Modified RK			RKF45		
	NSTEP	NFCN	RELERR	NSTEP	NFCN	RELERR
10^{-1}	2	14	0.60×10^{-1}	2	13	0.42×10^{-4}
10^{-2}	2	14	0.66×10^{-2}	2	13	0.74×10^{-4}
10^{-3}	3	21	0.54×10^{-3}	2	13	0.29×10^{-3}
10^{-4}	4	28	0.50×10^{-4}	3	19	0.14×10^{-4}
10^{-5}	5	35	0.47×10^{-5}	4	25	0.28×10^{-5}
10^{-6}	7	49	0.57×10^{-6}	6	37	0.30×10^{-6}
10^{-7}	9	63	0.99×10^{-7}	9	55	0.34×10^{-7}
10^{-8}	11	77	0.31×10^{-7}	13	79	0.37×10^{-8}
10^{-9}	16	112	0.67×10^{-8}	20	121	0.38×10^{-9}
10^{-10}	24	168	0.14×10^{-8}	31	187	0.39×10^{-10}

Table 4.2 Numerical results for the non-linear problem

EPS	Modified RK			RKF45		
	NSTEP	NFCN	ABSERR	NSTEP	NFCN	ABSERR
10^{-1}	6	42	0.35	8	64	0.24
10^{-2}	9	63	0.44×10^{-1}	10	76	0.47×10^{-1}
10^{-3}	12	84	0.70×10^{-2}	13	99	0.50×10^{-2}
10^{-4}	17	119	0.72×10^{-3}	16	107	0.14×10^{-2}
10^{-5}	22	154	0.15×10^{-3}	24	170	0.39×10^{-4}
10^{-6}	35	245	0.10×10^{-4}	35	231	0.28×10^{-5}
10^{-7}	50	350	0.22×10^{-5}	55	361	0.19×10^{-6}
10^{-8}	77	539	0.40×10^{-6}	85	546	0.14×10^{-7}
10^{-9}	115	805	0.48×10^{-7}	132	823	0.20×10^{-8}
10^{-10}	176	1232	0.69×10^{-8}	208	1284	0.18×10^{-9}

took fewer steps to complete the integration. It is hoped that in future research, something can be done to strengthen this weakness. The linear problem did not cause any trouble for both methods, so the modified RK method is more expensive to use since the cost of function evaluation is 7 per step compared to 6 per step for RKF45. However, in the non-linear problem where the behavior of the solution is not so predictable, the additional function evaluation paid off as RKF45 failed in many steps and had to retry several times.

LIST OF REFERENCES

[1] Bettis, D. G., Embedded Runge-Kutta methods of order four and five, Texas Institute for Computational Mechanics Report 78-2, University of Texas at Austin, Texas, 1978.

[2] Fehlberg, E., Klassiche Runge-Kutta formeln vierter und niedrigerer ordnung mit schrittweiten-controlle und inre anwendung auf warmeleitungsprobleme, Computing 6, 1970, 61-71.

[3] Gear, C. W., Runge-Kutta starters for multistep methods, ACM Transaction on Mathematical Software 6, 1980, 263-279.

[4] Gear, C. W., Runge-Kutta starters for multistep methods, Report 78-938, Department of Computer Science, University of Illinois at Urbana-Champaign, Illinois, 1978.

[5] Gear, C. W., Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, Englewood Cliffs, New Jersey, 1971.