Publ. RIMS, Kyoto Univ.
8 (1972/73), 13-42

A Geometric Method for the Numerical Solution of Nonlinear Equations and Its Application to Nonlinear Oscillations

By

Yoshitane SHINOHARA*

1. Introduction

In his previous paper [1, 2], the author proposed a new method for the numerical solution of a system of nonlinear equations

$$\text{(E)} \qquad \boldsymbol{F}(\boldsymbol{x})=\{f_k(x_1, x_2, \cdots, x_m)\}=\boldsymbol{0} \qquad (k=1, 2, \cdots, m)$$

in a bounded region. But in these papers he did not describe the techniques of programming for the method proposed.

In the present paper, the techniques of programming will be described and a program written in FORTRAN will be presented. This program is tested on a system consisting of five algebraic equations. Lastly, in illustration, will be shown an application of our program to the computation of subharmonic solutions of Duffing's equation.

The author expresses his hearty gratitude to Professor M. Urabe for his constant advice and encouragement.

The author also wishes to acknowledge the assistance of Miss. T. Takahashi, Mr. Y. Kono and Mr. Y. Nakatani in the computer room of the RIMS, Kyoto University.

Received September 29, 1971 and in revised form February 25, 1972.
Communicated by M. Urabe.
* Faculty of Engineering, Tokushima University, Tokushima 770, Japan.

2. The Method of Computation

We consider a global problem of finding the solutions of the system (E) in a bounded region R:

$$R=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \qquad (i=1, 2, \cdots, m)\}.$$

We assume that the real-valued functions $f_k(x_1, x_2, \cdots, x_m)$ $(k=1, 2, \cdots, m)$ are continuously differentiable in the region R. We assume further that the solutions under consideration are all simple, that is, for all solutions of (E) lying in R, the Jacobian of $\boldsymbol{F}(\boldsymbol{x})$ with respect to $\boldsymbol{x}$ does not vanish.

From the equations of the system (E), we choose $m-1$ equations

$$f_\alpha(x_1, x_2, \cdots, x_m)=0 \qquad (\alpha=1, 2, \cdots, m-1). \tag{2.1}$$

For this system, we request that the rank of the matrix

$$(\partial f_\alpha/\partial x_i) \qquad (\alpha=1, 2, \cdots, m-1;\ i=1, 2, \cdots, m)$$

is equal to $m-1$. The system of equations (2.1) then determines a curve

$$C: \quad \boldsymbol{x}=\boldsymbol{x}(s),$$

for which from (2.1) we have

$$\sum_{i=1}^{m} \frac{\partial f_\alpha}{\partial x_i} \cdot \frac{dx_i}{ds}=0 \qquad (\alpha=1, 2, \cdots, m-1). \tag{2.2}$$

Put

$$D_i=(-1)^i \cdot \frac{\partial(f_1, f_2, \cdots, f_{m-1})}{\partial(x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_m)} \qquad (i=1, 2, \cdots, m), \tag{2.3}$$

then from (2.2) we have

$$\frac{dx_i}{ds}=\lambda \cdot D_i \qquad (i=1, 2, \cdots, m), \tag{2.4}$$

where λ is an arbitrary parameter. Let us choose a parameter s so that

s may be an arc length of the curve C. Then we readily have

$$\lambda = \pm \left[\sum_{i=1}^{m} D_i^2 \right]^{-\frac{1}{2}}. \tag{2.5}$$

Hence from (2.4), for the curve C we have a system of differential equations of the form

$$\frac{d\boldsymbol{x}}{ds} = \boldsymbol{X}(\boldsymbol{x}). \tag{2.6}$$

Now we take a point $\boldsymbol{x} = \boldsymbol{x}^{(0)}$ on the curve C and suppose $\boldsymbol{x}^{(0)} = \boldsymbol{x}(0)$. Then we can trace the curve C integrating numerically equation (2.6) by a step-by-step method, say, the Runge-Kutta method. Let $\boldsymbol{x}^{(l)}$ $(l = 1, 2, \cdots)$ be an approximate value of $\boldsymbol{x}(s)$ obtained at the l-th step by the numerical integration. Then we may have $f_m[\boldsymbol{x}^{(0)}] \cdot f_m[\boldsymbol{x}^{(1)}] \leqq 0$. Otherwise we continue the numerical integration of (2.6) until we have

$$f_m[\boldsymbol{x}^{(l-1)}] \cdot f_m[\boldsymbol{x}^{(l)}] \leqq 0. \tag{2.7}$$

Once we have had (2.7) for some l, we check whether $|f_m[\boldsymbol{x}^{(l-1)}]|$ or $|f_m[\boldsymbol{x}^{(l)}]|$ is smaller than a specified positive number ε. If this is not satisfied, we multiply the step-size of the numerical integration by 2^{-p} $(p \geqq 1)$ and repeat this process. Then after a finite number of repetitions we shall have

$$\begin{cases} f_m[\boldsymbol{x}^{(l-1)}] \cdot f_m[\boldsymbol{x}^{(l)}] \leqq 0, \\ |f_m[\boldsymbol{x}^{(l-1)}]| \quad \text{or} \quad |f_m[\boldsymbol{x}^{(l)}]| < \varepsilon. \end{cases} \tag{2.8}$$

The value $\boldsymbol{x}^{(l-1)}$ or $\boldsymbol{x}^{(l)}$ satisfying (2.8) gives an approximate solution of the given system of equations (E). Starting from $\boldsymbol{x}^{(l-1)}$ or $\boldsymbol{x}^{(l)}$, we then can compute a solution of (E) by the Newton method. However, if ε *is* very small, $\boldsymbol{x}^{(l-1)}$ or $\boldsymbol{x}^{(l)}$ itself will give an accurate approximate solution of (E).

Our method is based on the above principle.

In order to find a point $\boldsymbol{x} = \boldsymbol{x}^{(0)}$ on the curve C, it suffices to find a solution of the system (2.1) consisting of $m-1$ equations after assigning a suitable value to some one of x_i's $(i=1, 2, \cdots, m)$. Our method is clearly applicable to systems consisting of $m-1$ equations. Hence, repeating such a process, our method is reduced to finding a solution of a single equation.

In the course of the numerical integration of (2.6), it may happen due to the accumulated error that

$$(2.9) \qquad |f_\alpha[\boldsymbol{x}^{(l)}]| \geqq \xi$$

at some l-th step for some positive integer $\alpha \leqq m-1$, where ξ is a prescribed positive number. When (2.9) happens, one however can correct $\boldsymbol{x}^{(l)}$ so that the corrected value $\tilde{\boldsymbol{x}}^{(l)}$ may satisfy inequalities

$$|f_\alpha[\tilde{\boldsymbol{x}}^{(l)}]| < \xi$$

for all $\alpha=1, 2, \cdots, m-1$. To do so, it suffices to apply the Newton method to (2.1) starting from $\boldsymbol{x}=\boldsymbol{x}^{(l)}$ leaving one of $x_i^{(e)}$'s $(i=1, 2, \cdots, m)$ unchanged.

If we continue the numerical integration of (2.6) beyond the approximate solution obtained or begin the numerical integration of (2.6) in the reverse direction, then we shall have (2.8) again provided there are solutions of (E) on the curve. Continuing our process, in the region R, we thus can get numerically all solutions of (E) lying on a branch of the curve C.

The curve C may consist of some different disconnected branches [Fig. 1, Fig. 2]. In order to find all these branches, a special setup is needed.

We divide each interval $[l_j, m_j]$ $(j=2, 3, \cdots, m)$ into subintervals of equal length h_j and denote an arbitrary subinterval obtained by $[L_j, L_j+h_j]$.

To begin with, we consider two equations

$$(2.10) \qquad f_1(x_1, x_2, l_3, L_4, \cdots, L_m)=0,$$

$$(2.11) \qquad f_2(x_1, x_2, l_3, L_4, \cdots, L_m)=0$$

in the region $R_1^{[2]}$:

$$R_1^{[2]}=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \ (i=1, 2), \ x_3=l_3, \\ x_j=L_j \ (j=4, 5, \cdots, m)\}.$$

Our method begins with the calculation of the intersections of two plane curves (2.10) and (2.11). In order to find all the branches of the curve (2.10) lying in the region $R_1^{[2]}$, we divide the region $R_1^{[2]}$ into subregions $D_i^{[2]}$ so that

$$R_1^{[2]}=D_1^{[2]}\cup D_2^{[2]}\cup\cdots\cup D_{n_2}^{[2]} \qquad \text{[Fig. 1]},$$

where $D_i^{[2]}$ are rectangles with the length m_1-l_1 and the breadth h_2.

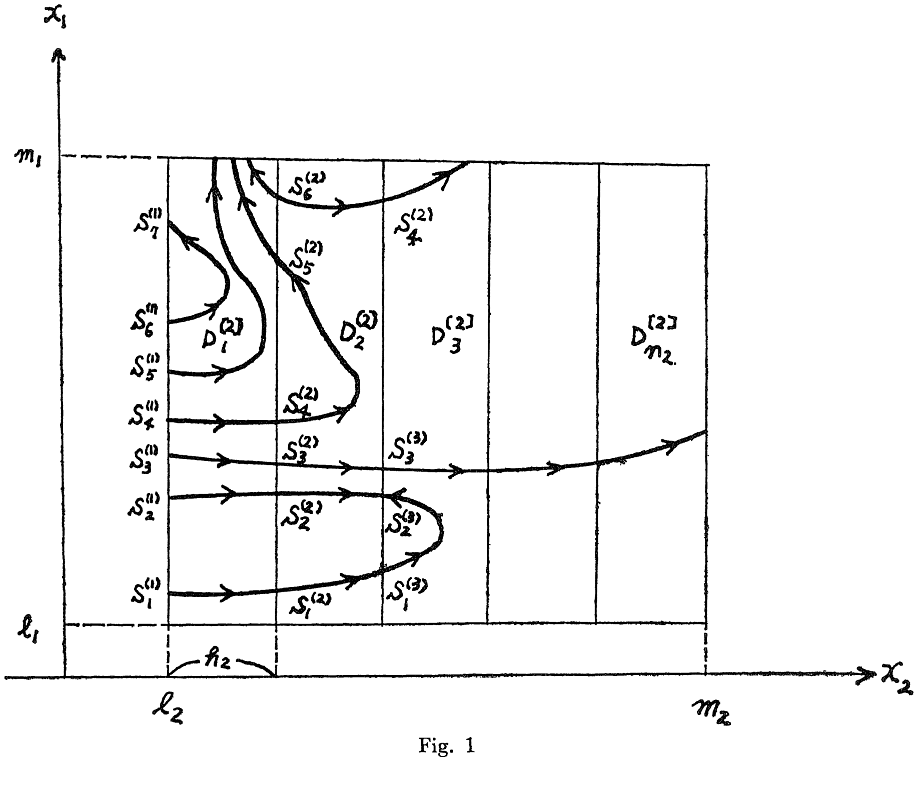

Fig. 1

First, we calculate the intersections $S_1^{(1)}, S_2^{(1)}, \cdots$ of the curve (2.10) with the straight line $x_2=l_2$. Starting from these points, we trace the branches of the curve (2.10) numerically until we reach the boundary of $D_1^{[2]}$. Suppose that the branches of the curve (2.10) have reached the boundary $x_2=l_2+h_2$ of $D_1^{[2]}$ at $S_1^{(2)}, S_2^{(2)}, \cdots$. Then we store these $S_1^{(2)}$, $S_2^{(2)}, \cdots$ as points of delivery for the succeeding step. In the course of tracing the branches of the curve (2.10), we compute the intersections of these branches with the curve (2.11) with sufficient accuracy.

Next, we calculate the intersections of the curve (2.10) with the straight line $x_2=l_2+h_2$. Some of them are points of delivery obtained at the preceding step. Starting from these points of delivery in the direction of increasing x_2, we trace the branches of the curve (2.10) numerically until we reach the boundary of $D_2^{[2]}$. Suppose that the branches of the curve (2.10) have reached the boundary $x_2=l_2+2h_2$ of $D_2^{[2]}$ at $S_1^{(3)}, S_2^{(3)}, \cdots$. Then we store these $S_1^{(3)}, S_2^{(3)}, \cdots$ as points of delivery for the succeeding step. In this tracing, it may happen that the branches starting from the points of delivery on the straight line $x_2=l_2+h_2$ return back to the points, say $S_5^{(2)}, \cdots$, on the straight line $x_2=l_2+h_2$ without going outside $D_2^{[2]}$. In such a case, the points $S_5^{(2)}, \cdots$ may be points of delivery obtained at the preceding step. In such a case the points $S_5^{(2)}, \cdots$ are called points of trivial delivery. When the points $S_5^{(2)}, \cdots$ are not points of delivery obtained at the preceding step, they are called points of reverse delivery. Clearly points of trivial delivery appear in pairs and we need not trace the branches of the curve (2.10) starting from both points in pairs. As the points of reverse delivery are concerned, it is necessary to trace the branches of the curve (2.10) in the direction of decreasing x_2 starting from these points. Among the intersections of the curve (2.10) with the straight line $x_2=l_2+h_2$, there may be some points, say $S_6^{(2)}, \cdots$, which are neither points of delivery nor points of reverse delivery. When such points appear, it is necessary to trace the branches of the curve (2.10) starting from these points in two directions, that is, the direction of increasing x_2 and that of decreasing x_2. When we trace the branches of the curve (2.10) in the direction of increasing x_2, we may reach the straight line $x_2=l_2+2h_2$ at some points. In such a case, we store these points as points of delivery for the succeeding step. In the course of tracing the branches of the curve (2.10), we always compute the intersections of these branches with the curve (2.11) with sufficient accuracy.

We continue the above process step by step. Then we shall obtain all solutions of the simultaneous equations (2.10) and (2.11) lying in the region $R_1^{[2]}=D_1^{[2]}\cup D_2^{[2]}\cup\cdots\cup D_{n_2}^{[2]}$.

Let $S_1^{(1)}$, $S_2^{(1)}$, $S_3^{(1)}$, $S_4^{(1)}$, $\cdots$ be the points corresponding to the solutions of the simultaneous equations (2.10) and (2.11) lying in $R_1^{[2]}$ (Fig. 2).

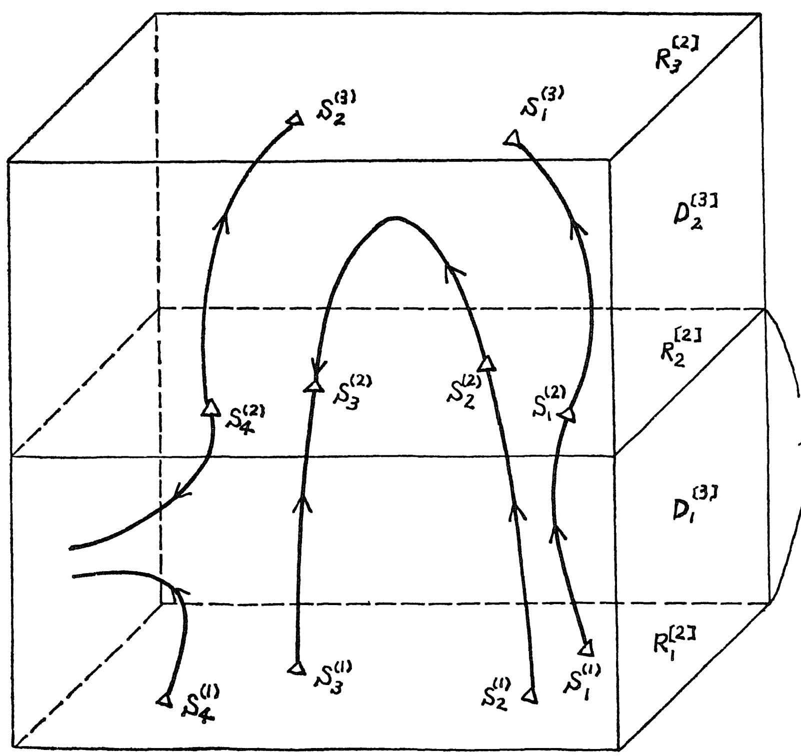

Fig. 2

We consider the three equations

(2.12) $$f_1(x_1, x_2, x_3, L_4, \cdots, L_m)=0,$$

(2.13) $$f_2(x_1, x_2, x_3, L_4, \cdots, L_m)=0,$$

(2.14) $$f_3(x_1, x_2, x_3, L_4, \cdots, L_m)=0$$

in the region $D_1^{[3]}$:

$$D_1^{[3]}=\{(x_1, x_2, x_3, x_4, \cdots, x_m): l_i \leqq x_i \leqq m_i \quad (i=1, 2),$$
$$l_3 \leqq x_3 \leqq l_3+h_3,\ x_j=L_j \quad (j=4, 5, \cdots, m)\}.$$

Starting from the points $S_1^{(1)}, S_2^{(1)}, S_3^{(1)}, S_4^{(1)}, \cdots$, we trace the branches of the curve defined by (2.12) and (2.13) numerically until we reach the boundary of $D_1^{[3]}$. If these branches reach the boundary $x_3=l_3+h_3$ of $D_1^{[3]}$, say at $S_1^{(2)}, S_2^{(2)}, S_3^{(2)}, \cdots$, then we store these $S_1^{(2)}, S_2^{(2)}, S_3^{(2)}, \cdots$ as points of delivery for the succeeding step. In the course of tracing the branches of the curve defined by (2.12) and (2.13), we compute the intersections of these branches with the surface (2.14) with sufficient accuracy.

Next, by the use of the procedure applied to the simultaneous equations (2.10) and (2.11), we calculate the solutions of the simultaneous equations

$$\text{(2.15)} \qquad \begin{cases} f_1(x_1, x_2, l_3+h_3, L_4, \cdots, L_m)=0, \\ f_2(x_1, x_2, l_3+h_3, L_4, \cdots, L_m)=0 \end{cases}$$

in the region $R_2^{[2]}$:

$$R_2^{[2]}=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \quad (i=1, 2),$$
$$x_3=l_3+h_3, \quad x_j=L_j \quad (j=4, 5, \cdots, m)\}.$$

Some of the solutions correspond to points of delivery obtained at the preceding step. Starting from these points of delivery in the direction of increasing x_3, we trace the branches of the curve defined by (2.12) and (2.13) numerically until we reach the boundary of $D_2^{[3]}$:

$$D_2^{[3]}=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \quad (i=1, 2),$$
$$l_3+h_3 \leqq x_3 \leqq l_3+2h_3, \quad x_j=L_j \quad (j=4, 5, \cdots, m)\}.$$

If these branches reach the boundary $x_3=l_3+2h_3$ of $D_2^{[3]}$, say at $S_1^{(3)}, \cdots$, then we store these $S_1^{(3)}, \cdots$ as points of delivery for the succeeding step.

If the branches starting from the points of delivery on the plane $x_3 = l_3 + h_3$ return back to the points, say $S_3^{(2)}, \cdots,$ on the plane $x_3 = l_3 + h_3$ without going outside $D_2^{[3]}$, then the points $S_3^{(2)}, \cdots$ are either points of trivial delivery, that is, the points coincident with some points of delivery obtained at the preceding step, or points of reverse delivery, that is, the points which do not coincide with any points of delivery obtained at the preceding step. Clearly points of trivial delivery appear in pairs and we need not trace the branches of the curve defined by (2.12) and (2.13) from both points in pairs. From the points of reverse delivery, however, it is necessary to trace the branches of the curve defined by (2.12) and (2.13) in the direction of decreasing x_3. Among the points corresponding to the solutions of (2.15), there may be some points, say $S_4^{(2)}, \cdots,$ which are neither points of delivery nor points of reverse delivery. From these points, if exist, it is necessary to trace the branches of the curve defined by (2.12) and (2.13) in two directions, that is, the direction of increasing x_3 and that of decreasing x_3. When we trace the branches of the curve defined by (2.12) and (2.13) in the direction of increasing x_3, we may reach the plane $x_3 = l_3 + 2h_3$ at some points. In such a case, we store these points as points of delivery for the succeeding step. In the course of tracing the branches of the curve defined by (2.12) and (2.13), we always compute the intersections of these branches with the surface (2.14) with sufficient accuracy.

We continue the above process step by step. Then we shall obtain all solutions of the simultaneous equations (2.12), (2.13) and (2.14) lying in the region $R_1^{[3]} = D_1^{[3]} \cup D_2^{[3]} \cup \cdots \cup D_{n_3}^{[3]}$.

Let $S_1^{(1)}, S_2^{(1)}, \cdots$ be the points corresponding to the solutions (2.12), (2.13) and (2.14) lying in $R_1^{[3]}$. We consider the four equations

(2.16) $$f_1(x_1, x_2, x_3, x_4, L_5, \cdots, L_m) = 0,$$

(2.17) $$f_2(x_1, x_2, x_3, x_4, L_5, \cdots, L_m) = 0,$$

(2.18) $$f_3(x_1, x_2, x_3, x_4, L_5, \cdots, L_m) = 0,$$

(2.19) $$f_4(x_1, x_2, x_3, x_4, L_5, \cdots, L_m) = 0$$

in the region $D_1^{[4]}$:

$$D_1^{[4]}=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \quad (i=1, 2, 3),$$

$$l_4 \leqq x_4 \leqq l_4+h_4, \quad x_j=L_j \quad (j=5, 6, \cdots, m)\}.$$

Starting from the points $S_1^{(1)}, S_2^{(1)}, \cdots$, we trace the branches of the curve defined by (2.16), (2.17) and (2.18) numerically until we reach the boundary of $D_1^{[4]}$. Then we repeat the process applied for the simultaneous equations (2.12), (2.13) and (2.14). Then we shall obtain all solutions of the simultaneous equations (2.16), (2.17), (2.18) and (2.19) lying in the region $R_1^{[4]}=D_1^{[4]} \cup D_2^{[4]} \cup D_3^{[4]} \cup \cdots \cup D_{n_4}^{[4]}$, where

$$D_k^{[4]}=\{(x_1, x_2, \cdots, x_m): l_i \leqq x_i \leqq m_i \quad (i=1, 2, 3),$$

$$l_4+(k-1)h_4 \leqq x_4 \leqq l_4+kh_4, \quad x_j=L_j \quad (j=5, 6, \cdots, m)\}.$$

Continuing the above process, after a finite number of steps, we obtain all solutions of the given system (E) lying in the region $R=R_1^{[m]}$.

3. Program and Its Test

We have written a program of our method in FORTRAN and have tested it on a system of algebraic equations which appears in the factorization of a polynomial.

Our system of algebraic equations is connected with the factorization of the polynomial

$$P(x)=x^7+a_1 x^6+a_2 x^5+\cdots+a_6 x+a_7$$

into

$$(x^5+px^4+qx^3+rx^2+sx+t)\cdot(x^2+lx+m).$$

Put

$$P(x)=(x^5+px^4+qx^3+rx^2+sx+t)\cdot(x^2+lx+m)$$

$$-(k_1 x^4+k_2 x^3+k_3 x^2+k_4 x+k_5),$$

then comparing the coefficients of powers of x in both sides, we have

$$\begin{cases} p+l=a_1, \\ q+pl+m=a_2, \\ r+ql+pm-k_1=a_3, \\ s+rl+qm-k_2=a_4, \\ t+sl+rm-k_3=a_5, \\ tl+sm-k_4=a_6, \\ tm-k_5=a_7. \end{cases}$$

Eliminating l and m from these equations, we have

$$\text{(3.1)} \quad \begin{cases} k_1=f_1(p, q, \cdots, t)=p^3-2pq+r-a_1\cdot(p^2-q)+a_2p-a_3, \\ k_2=f_2(p, q, \cdots, t)=p^2q-q^2-pr+s-a_1\cdot(pq-r)+a_2q-a_4, \\ k_3=f_3(p, q, \cdots, t)=p^2r-ps-qr+t-a_1\cdot(pr-s)+a_2r-a_5, \\ k_4=f_4(p, q, \cdots, t)=p^2s-pt-qs-a_1\cdot(ps-t)+a_2s-a_6, \\ k_5=f_5(p, q, \cdots, t)=p^2t-qt-a_1pt+a_2t-a_7. \end{cases}$$

Hence we see that $Q(x)=x^5+px^4+qx^3+rx^2+sx+t$ is a factor of the given polynomial $P(x)$ if and only if the coefficients (p, q, r, s, t) of $Q(x)$ satisfy the equations

$$\text{(3.2)} \qquad f_k(p, q, r, s, t)=0 \qquad (k=1, 2, 3, 4, 5).$$

We have tested our program on the system of equations (3.2) for the polynomial

$$\text{(3.3)} \qquad P(x)=(x^2+1)\cdot(x^2+x+1)\cdot(x+\alpha)\cdot(x+\beta)\cdot(x+\gamma).$$

For (3.3),

$$\text{(3.4)} \quad \begin{cases} a_1=1+A,\ a_2=2+A+B,\ a_3=1+2A+B+C, \\ a_4=1+A+2B+C,\ \ a_5=A+B+2C,\ \ a_6=B+C,\ \ a_7=C, \end{cases}$$

where

$$(3.5)\qquad \begin{cases} A=\alpha+\beta+\gamma, \\ B=\alpha\beta+\beta\gamma+\gamma\alpha, \\ C=\alpha\beta\gamma. \end{cases}$$

For $\alpha=0.5$, $\beta=-0.5$, $\gamma=-1$, we have tested our program. In this case, by (3.5),

$$A=-1,\quad B=-0.25,\quad C=0.25,$$

therefore, by (3.4), we have

$$a_1=0,\quad a_2=0.75,\quad a_3=-1,\quad a_4=-0.25,$$

$$a_5=-0.75,\quad a_6=0,\quad a_7=0.25.$$

Hence, by (3.1), the equations (3.2) become

$$(3.6)\qquad \begin{cases} f_1(p, q, r, s, t)=p^3-2pq+r+0.75p+1=0, \\ f_2(p, q, r, s, t)=p^2q-q^2-pr+s+0.75q+0.25=0, \\ f_3(p, q, r, s, t)=p^2r-ps-qr+t+0.75r+0.75=0, \\ f_4(p, q, r, s, t)=p^2s-pt-qs+0.75s=0, \\ f_5(p, q, r, s, t)=p^2t-qt+0.75t-0.25=0. \end{cases}$$

Now $Q(x)$ is a factor of $P(x)$ given by (3.3). Therefore, if p, q, r, s and t are all real, then $Q(x)$ must be one of the following polynomials:

$$(x^2+1)\cdot(x+0.5)\cdot(x-0.5)\cdot(x-1)$$

$$=x^5-x^4+0.75x^3-0.75x^2-0.25x+0.25,$$

$$(x^2+x+1)\cdot(x+0.5)\cdot(x-0.5)\cdot(x-1)=x^5-0.25x^3-x^2+0.25,$$

$$(x^2+1)\cdot(x^2+x+1)\cdot(x+0.5)=x^5+1.5x^4+2.5x^3+2x^2+1.5x+0.5,$$

$$(x^2+1)\cdot(x^2+x+1)\cdot(x-0.5)=x^5+0.5x^4+1.5x^3+0.5x-0.5,$$

$$(x^2+1)\cdot(x^2+x+1)\cdot(x-1)=x^5+x^3-x^2-1.$$

This means that the real solutions of equation (3.6) are

$$(p, q, r, s, t)=\begin{cases}(-1, 0.75, -0.75, -0.25, 0.25),\\(0, -0.25, -1, 0, 0.25),\\(1.5, 2.5, 2, 1.5, 0.5),\\(0.5, 1.5, 0, 0.5, -0.5),\\(0, 1, -1, 0, -1).\end{cases}$$

By the use of our program, we have computed real solutions of the equations (3.6) in the region:

$$-1.5 \leqq p \leqq 2, \quad -0.6 \leqq q \leqq 3, \quad -1.5 \leqq r \leqq 2.5,$$

$$-0.5 \leqq s \leqq 1.9, \quad -1 \leqq t \leqq 1.$$

The result obtained is as follows.

p	$-0.10000\ 00000$	$0.53118\ 62061 \times 10^{-12}$	$1.50000\ 0000$
q	$0.75000\ 00000$	$-0.25000\ 00000$	$2.50000\ 0000$
r	$-0.75000\ 00000$	$-1.00000\ 0000$	$2.00000\ 0000$
s	$-0.25000\ 00000$	$-0.80144\ 22459 \times 10^{-15}$	$1.50000\ 0000$
t	$0.25000\ 00000$	$0.25000\ 00000$	$0.50000\ 00000$

p	$0.50000\ 00000$	$0.19005\ 17937 \times 10^{-11}$
q	$1.50000\ 0000$	$1.00000\ 0000$
r	$-0.51261\ 91138 \times 10^{-11}$	$-1.00000\ 0000$
s	$0.50000\ 00000$	$0.81179\ 75042 \times 10^{-11}$
t	$-0.50000\ 00000$	$-1.00000\ 0000$

This result shows the usefulness of our method and program.

4. Application to the Computation of Subharmonic Solutions of Duffing's Equation

In papers [3] and [5], M. Urabe has developed Galerkin's procedure for the computation of periodic solutions of nonlinear periodic differential equations, and in paper [4], he has investigated the subharmonic solutions of Duffing's equation by means of the techniques developed by himself in [3] and [5]. In his method, the Newton method is employed for the numerical solution of the determining equation, that is, the equation which should be satisfied by the Fourier coefficients of the trigonometric polynomial approximating the desired periodic solution. In order to find starting approximate solutions of the determining equation, he solves the determining equation consisting of a very small number of equations making use of graphical methods or perturbation techniques. This method of finding starting approximate solutions, however, does not seem to work well always.

In order to find starting approximate solutions, we can apply our method to the determining equation consisting of a moderate number of equations. For experimentation, we have applied our method to the determining equation for 1/3-order subharmonic solutions of Duffing's equation.

We consider Duffing's equation in the form

$$\frac{d^2x}{dt^2}+\frac{\sigma}{\omega}\cdot\frac{dx}{dt}+\frac{1}{\Omega}\cdot x(1+\varepsilon x^2)=\frac{1}{\Omega}\cdot\cos t, \tag{4.1}$$

where $\Omega=\omega^2$.

A 1/3-order subharmonic solution of (4.1) is a solution of (4.1) of the form

$$x(t)=c_1+\sum_{n=1}^{\infty}\left(c_{2n}\cdot\sin\frac{n}{3}t+c_{2n+1}\cdot\cos\frac{n}{3}t\right). \tag{4.2}$$

Replacing t by $3t$ in (4.1) and (4.2), we see that a 1/3-order subharmonic solution of (4.1) is a solution of the form

$$x(t)=c_1+\sum_{n=1}^{\infty}(c_{2n}\cdot\sin nt+c_{2n+1}\cdot\cos nt) \tag{4.3}$$

of the equation

$$\frac{d^2x}{dt^2}+\frac{3\sigma}{\omega}\cdot\frac{dx}{dt}+\frac{9}{\Omega}\cdot x(1+\varepsilon x^2)=\frac{9}{\Omega}\cdot\cos 3t. \tag{4.4}$$

If $x(t)$ is a solution of (4.4), then $-x(t+\pi)$ is also a solution of (4.4). Hence we suppose that

$$-x(t+\pi)=x(t)$$

is valid for every 1/3-order subharmonic solution (this is certified for subharmonic solutions obtained in [4] by numerical computations). Then for a 1/3-order subharmonic solution, instead of (4.3), we have

$$x(t)=\sum_{n=1}^{\infty}[c_{2n}\cdot\sin(2n-1)t+c_{2n+1}\cdot\cos(2n-1)t]. \tag{4.5}$$

In order to find starting approximate solutions, we consider a Galerkin approximation (see [4]) of the form

$$\bar{x}(t)=c_2\cdot\sin t+c_3\cdot\cos t+c_4\cdot\sin 3t+c_5\cdot\cos 3t. \tag{4.6}$$

By [4], we have the following determining equation for (4.6):

$$\begin{cases} f_1(p,q,r,s)=\left(\dfrac{9}{\Omega}-1\right)p-\dfrac{3\sigma}{\omega}q+\dfrac{9\varepsilon}{\Omega}\cdot(0.75p^3-0.75p^2r \\ \qquad +0.75q^2r+0.75pq^2+1.5pr^2+1.5ps^2-1.5pqs)=0, \\ f_2(p,q,r,s)=\dfrac{3\sigma}{\omega}\cdot p+\left(\dfrac{9}{\Omega}-1\right)q+\dfrac{9\varepsilon}{\Omega}\cdot(0.75q^3+0.75p^2q \\ \qquad -0.75p^2s+0.75q^2s+1.5qr^2+1.5qs^2+1.5pqr)=0, \\ f_3(p,q,r,s)=\left(\dfrac{9}{\Omega}-9\right)r-\dfrac{9\sigma}{\omega}s+\dfrac{9\varepsilon}{\Omega}\cdot(-0.25p^3+0.75r^3 \\ \qquad +1.5p^2r+1.5q^2r+0.75pq^2+0.75rs^2)=0, \\ f_4(p,q,r,s)=\dfrac{9\sigma}{\omega}\cdot r+\left(\dfrac{9}{\Omega}-9\right)s-\dfrac{9}{\Omega}+\dfrac{9\varepsilon}{\Omega}\cdot(0.25q^3+0.75s^3 \\ \qquad -0.75p^2q+1.5p^2s+1.5q^2s+0.75r^2s)=0, \end{cases} \tag{4.7}$$

where

$$p=c_2, \quad q=c_3, \quad r=c_4, \quad s=c_5.$$

For

$$\sigma=2^{-5}, \quad \varepsilon=1, \quad \omega=4, \tag{4.8}$$

by the use of our program we have computed solutions of (4.7) in the region:

$$|p|\leqq 3, \quad |q|\leqq 3, \quad |r|\leqq 0.3, \quad |s|\leqq 0.3,$$

and we have obtained the following solutions.

	p	q	r	s
1:	0.72425 89710	−0.73255 43253	0.01522 20003	−0.06028 79583
2:	0.27228 11702	0.99350 38304	0.01522 20003	−0.06028 79583
3:	−0.99654 01409	−0.26094 95049	0.01522 20003	−0.06028 79583
4:	0.66808 50948	0.71625 13275	0.01424 33206	−0.08455 08252
5:	−0.95433 43925	0.22045 30000	0.01424 33206	−0.08455 08252
6:	0.28624 92976	−0.93670 43277	0.01424 33206	−0.08455 08252
7:	0.00000 00000	0.00000 00000	0.00055 57640	−0.06667 68579

Table 1

Now from the form of (4.4), we can easily see that if $x(t)$ is a solution of (4.4), then $x[t+(2\pi/3)]$ and $x[t+(4\pi/3)]$ are also solutions of (4.4), and that if $\bar{x}(t)$ is a Galerkin approximation of a 2π-periodic solution of (4.4), then $\bar{x}[t+(2\pi/3)]$ and $\bar{x}[t+(4\pi/3)]$ are also Galerkin approximations of 2π-periodic solutions of (4.4) with the same order as $\bar{x}(t)$. For the solutions of (4.7) shown in Table 1, we readily see that the Galerkin approximations $\bar{x}_2(t)$ and $\bar{x}_3(t)$ corresponding to the 2nd and 3rd solutions in Table 1 are equal respectively to $\bar{x}_1[t+(2\pi/3)]$ and $\bar{x}_1[t+(4\pi/3)]$, where $\bar{x}_1(t)$ is the Galerkin approximation corresponding to the first solution in Table 1. Likewise we readily see that the Galerkin approximations $\bar{x}_5(t)$ and $\bar{x}_6(t)$ corresponding to the 5th and 6th solutions in Table 1 are equal respectively to $\bar{x}_4[t+(2\pi/3)]$ and $\bar{x}_4[t+(4\pi/3)]$, where $\bar{x}_4(t)$ is the Galerkin approximation corresponding to the 4th solution in Table 1. The Galerkin approximation $\bar{x}_7(t)$ corresponding to the 7th solution in Table 1 will be supposed to be a Galerkin approximation of a harmonic solution of original Duffing's equation (4.1).

Starting from the solutions of (4.7) shown in Table 1, by the use of the techniques described in [4] we have computed Galerkin approximations of higher order for subharmonic solutions and harmonic solution. However, by the reason mentioned above, we have not carried out the computations starting from the 2nd, 3rd, 5th and 6th solutions in Table 1. Tables 2 and 3 show the results. In these tables, for each approximate solution, is given an error bound δ such that

$$[\,|\bar{x}(t)-\hat{x}(t)|^2+|\dot{\bar{x}}(t)-\dot{\hat{x}}(t)|^2]^{\frac{1}{2}}\leqq\delta,$$

where $\cdot = d/dt$ and $\hat{x}(t)$ is an exact solution corresponding to the approximate solution $\bar{x}(t)$.

Periodic solutions of (4.4) with $\sigma=2^{-5}$, $\varepsilon=1$, $\omega=4$:

1 $\bar{x}_1(t)=0.72456\ 14343 \sin t \quad -0.73222\ 00674 \cos t$
$\quad +0.01522\ 23982 \sin 3t \quad -0.06033\ 11349 \cos 3t$
$\quad +0.00112\ 92234 \sin 5t \quad +0.00021\ 38735 \cos 5t$
$\quad +0.00003\ 31833 \sin 7t \quad -0.00000\ 00135 \cos 7t$
$\quad +0.00000\ 05831 \sin 9t \quad +0.00000\ 06017 \cos 9t$
$\quad +0.00000\ 00138 \sin 11t \quad +0.00000\ 00272 \cos 11t$
$\quad -0.00000\ 00001 \sin 13t \quad +0.00000\ 00007 \cos 13t.$

$\delta=6.6\times10^{-8}$, Stability: stable.

2 $\bar{x}_4(t)=0.66825\ 85789 \sin t \quad +0.71578\ 29204 \cos t$
$\quad +0.01424\ 01915 \sin 3t \quad -0.08465\ 09661 \cos 3t$
$\quad -0.00154\ 34867 \sin 5t \quad -0.00028\ 97473 \cos 5t$
$\quad +0.00002\ 33942 \sin 7t \quad +0.00007\ 35294 \cos 7t$

$$
\begin{array}{llll}
+0.00000\ 22613 & \sin 9t & -0.00000\ 16730 & \cos 9t \\
-0.00000\ 00815 & \sin 11t & -0.00000\ 00660 & \cos 11t \\
-0.00000\ 00013 & \sin 13t & +0.00000\ 00037 & \cos 13t \\
+0.00000\ 00001 & \sin 15t. & &
\end{array}
$$

$\delta=1.3\times 10^{-7}$, Stability: unstable.

Table 2

Periodic solution to (4.1) with $\sigma=2^{-5}$, $\varepsilon=1$, $\omega=4$:

$$
\begin{aligned}
\bar{x}_7(t)=0.00055\ 57640 \sin t \quad & -0.06667\ 68581 \cos t \\
+0.00000\ 00143 \sin 3t \quad & -0.00000\ 05181 \cos 3t.
\end{aligned}
$$

$\delta=1.5\times 10^{-9}$, Stability: stable.

Table 3

References

[1] Shinohara, Y., A geometric method of numerical solution of nonlinear equations and error estimate by Urabe's proposition, *Publ. RIMS. Kyoto Univ.* **5** (1969), 1-9.

[2] ———, The geometric method and a generalized Bairstow method for numerical solution of polynomial equation, *J. Math. Tokushima Univ.* **4** (1970), 19-32.

[3] Urabe, M., Galerkin's procedure for nonlinear periodic systems, *Arch. Rational Mech. Anal.* **20** (1965), 120-152.

[4] ———, Numerical investigation of subharmonic solutions to Duffing's equation, *Publ. RIMS. Kyoto Univ.* **5** (1969), 79-112.

[5] Urabe, M. and A. Reiter, Numerical computation of nonlinear forced oscillations by Galerkin's procedure, *J. Math Appl.* **14** (1966), 107-140.

[6] Yamauchi, J., S. Moriguchi and S. Hitotumatu, Denshikeisanki no tame no suchikeisanho **1**, Baifukan, 1965 (Japanese).

```
C       MAIN PROGRAM FOR NUMERICAL SOLUTION OF NONLINEAR EQUATIONS
C
C       ANSX IS THE SOLUTION VECTOR.
C       THE NUMBER OF SOLUTIONS IS STORED IN NUM.
C
C       THE TWO FORMATS(NO. 4000,5000) ARE NECESSARY.
C       DIMENSION   X(MAX),F(MAX),FX(MAX,MAX),XM(MAX),XL(MAX),XH(MAX),
C                   DXM(MAX),DXL(MAX),ANSX(MAX,20,MAX)
C       COMMON EPSIL,F,FX,XM,XL,XH,DXM,DXL,ANSX
C
        DIMENSION X(5),F(5),FX(5,5),XM(5),XL(5),XH(5),DXM(5),DXL(5),
     +  ANSX(5,20,5)
        COMMON EPSIL,F,FX,XM,XL,XH,DXM,DXL,ANSX
        M=5
        READ(5,1000) (XM(I),XL(I),XH(I),I=1,M)
        READ(5,1000) (DXM(I),DXL(I),I=1,M)
        CALL GLOBAL(M,NUM)
        IF(NUM) 10,11,12
   10   WRITE(6,5000)
        GO TO 15
   11   WRITE(6,4000)
        GO TO 15
   12   WRITE(6,2000) NUM
        DO 14 K=1,NUM
        DO 13 I=1,M
   13   X(I)=ANSX(I,K,M)
        CALL KANSU(X,F,M)
   14   WRITE(6,3000) M,(X(LM),LM=1,M),(F(LM),LM=1,M)
 1000   FORMAT(15F5.2)
 2000   FORMAT(1H1,25H*****NUMBER OF SOLUTIONS=,I3)
 3000   FORMAT(1H ,3X,10HDIMENSION=,I3,(/5X,5E20.10))
 4000   FORMAT(1H ,18H*****NOT  OBTAINED)
 5000   FORMAT(1H ,22H***MAKE MEMORY GREATER)
   15   CALL EXIT
        END
C       SUBROUTINE GLOBAL(M,NUM)
C       GEOMETRIC METHOD FOR FIVE VARIABLES, MAY 25,1971,SHINOHARA
C       THIS SUBPROGRAM COMPUTES THE ZEROS OF A SYSTEM OF EQUATIONS
C       F(X)=0, WHERE F AND X ARE M-DIMENSIONAL VECTORS SUCH THAT
C       F=(F1,F2,...,FM) AND X=(X(1),X(2),...,X(M)), IN A BOUNDED
C       REGION R=(X(I), XL(I)≦X(I)≦XM(I), I=1,2,...,M).
C
C       INPUTING DATA ARE XL(I),XM(I) AND THE BREADTH XH(I) (I=1,2,
C       ...,M).
C       AUXILIARY DATA DXL(I),DXM(I) ARE ALSO NECESSARY, WHERE
C       DXL(I)≦XL(I), DXM(I)≧XM(I).
C
C       DIMENSION   X(MAX),F(MAX),FX(MAX,MAX),ANSX(MAX,20,MAX),
C                   IANS(MAX),KRANS(20,MAX),GG2(2),WX(MAX),RUNGE(MAX,4)
C                   ,YF(MAX,20),YYF(MAX,20),KS(MAX),KKS(MAX),KSR(20),
C                   KKR(20),XM(MAX),XL(MAX),XIL(MAX),XIR(MAX),XH(MAX),


C                   A(MAX,MAX),RUNGX(MAX),LLJ(MAX),MJ(MAX),DXM(MAX),
C                   DXL(MAX),MANS(KIZAMI), WHERE 20 DENOTES A SIZE OF
C                   THE MEMORY.
C       COMMON EPSIL,F,FX,XM,XL,XH,DXM,DXL,ANSX
C
C       THIS PROGRAM IS WRITTEN FOR THE CASE MAX=5.
C       THIS PROGRAM WORKS FOR M SUCH THAT MAX≧M≧2.
C
C       M IS THE NUMBER OF DIMENSION OF THE VECTOR X.
C       THE NUMBER OF SOLUTIONS IS STORED IN NUM.
C       ANSX IS THE SOLUTION VECTOR.
C
        SUBROUTINE GLOBAL(M,NUM)
```

```
      DIMENSION X(5),F(5),FX(5,5),ANSX(5,20,5),IANS(5),KRANS(20,5),
     1 GG2(2),WX(5),RUNGE(5,4),YF(5,20),YYF(5,20),KS(5),KKS(5),
     2 KSR(20),KKR(20),XM(5),XL(5),XIL(5),XIR(5),XH(5),A(5,5),RUNGX(5),
     3     LLJ(5),MJ(5),DXM(5),DXL(5),MANS(5)
      COMMON EPSIL,F,FX,XM,XL,XH,DXM,DXL,ANSX
      KIZAMI=5
C     NUMBER OF BITS OF MANTISSA
      MANT=37
      EPSIL=0.5**MANT*16.0
      CONST=100.0*EPSIL
C     NUMBER OF EFFECTS
      MEMORY=20
      DO 10 I=1,5
   10 MJ(I)=1
C     DOMAIN
      WRITE(6,1200) (XM(I),XL(I),XH(I),I=1,M)
      WRITE(6,1300) (DXM(I),DXL(I),I=1,M)
      DO 999  MCUT=1,KIZAMI
      WRITE(6,1400) EPSIL,CONST
      DO 11 I=1,M
   11 MJ(I)=(XM(I)-XL(I))/XH(I)+1.0
      DELTAS=0.03125
      EH=DELTAS
      ALPHA=0.1E-04
      JIGEN=2
      IANS(5)=0
      IANS(4)=0
C     FIVE-DIMENSIONAL CASE
      M5=MJ(5)
      M4=MJ(4)
      M3=MJ(3)
      M2=MJ(2)
      DO 500 LL5=1,M5
      LLJ(5)=LL5
      XIL(5)=XL(5)+FLOAT(LL5-1)*XH(5)
      XIR(5)=XIL(5)+XH(5)
      WALPHA=0.1E-04
      KS(4)=0
      KKS(5)=0
      X(5)=XIL(5)
      WX(5)=X(5)
      RUNGX(5)=X(5)
      IANS(3)=0
C     FOUR-DIMENSIONAL CASE
  757 DO 400 LL4=1,M4
      IF(JIGEN-2) 756,756,755
  756 LLJ(4)=LL4
      XIL(4)=XL(4)+FLOAT(LL4-1)*XH(4)
      XIR(4)=XIL(4)+XH(4)
      WALPHA=0.1E-04
      KS(3)=0
      KKS(4)=0
      X(4)=XIL(4)


      WX(4)=X(4)
      RUNGX(4)=X(4)
      IANS(2)=0
C     THREE-DIMENSIONAL CASE
  755 DO 300 LL3=1,M3
      IF(JIGEN-2) 753,753,754
  754 IF(M4O2) 218,203,218
  753 LLJ(3)=LL3
      XIL(3)=XL(3)+FLOAT(LL3-1)*XH(3)
      XIR(3)=XIL(3)+XH(3)
      WALPHA=0.1E-04
      KS(2)=0
```

```
      KKS(3)=0
      X(3)=XIL(3)
      WX(3)=X(3)
      RUNGX(3)=X(3)
C     TWO-DIMENSIONAL CASE
  752 DO 200 LL2=1,M2
      IF(JIGEN-2) 751,751,25
  751 LLJ(2)=LL2
      XIL(2)=XL(2)+FLOAT(LL2-1)*XH(2)
      XIR(2)=XIL(2)+XH(2)
      WALPHA=0.1E-04
      KKS(2)=0
      JIGEN=2
      IANS(1)=1
      YL=XL(1)
   12 X(2)=XIL(2)
C     DETERMINE X(1) SATISFYING F(X(1),XIL(2),...,XIL(5))=0
      CALL KUTTA(X,WX,EH,2,JIGEN,1,LYESNO,YL,XM(1),0)
      IF(LYESNO) 13,14,14
   13 CONTINUE
      IF(XM(2)-XIL(2)) 201,127,127
   14 YL=X(1)+SQRT(EPSIL)
   15 CALL KANSU(X,F,JIGEN)
      GG2(1)=F(2)
      JA=IANS(1)
      DO 16 LM=1,JIGEN
   16 ANSX(LM,JA,1)=X(LM)
      IF(KS(2)) 17,24,17
   17 JB=KS(2)
      GOSA=ALPHA*ABS(X(1))
      IF(GOSA-EPSIL) 18,18,19
   18 GOSA=EPSIL
   19 CONTINUE
      DO 23 LM=1,JB
      IF(ABS(YF(1,LM)-X(1))-GOSA) 20,20,23
   20 IF(KSR(LM)-2) 21,22,12
   21 KKRR=1
      GO TO 25
   22 KKRR=2
      GO TO 25
   23 CONTINUE
   24 KKRR=0


   25 CONTINUE
C     TRACE A BRANCH OF THE CURVE
      DO 100 KR=1,2
      IF(KKRR-2) 27,26,26
   26 KKRR=1
      GO TO 100
   27 JB=IANS(1)
      DO 28 LM=1,JIGEN
   28 X(LM)=ANSX(LM,JB,1)
      CALL KANSU(X,F,JIGEN)
      GG2(1)=F(JIGEN)
   29 H=DELTAS
      EH=H
   30 CALL BIBUN(X,FX,JIGEN)
C     TEST FOR SINGULAR POINT
      IF(JIGEN-2) 37,37,31
   31 IA=JIGEN-1
      DO 34 IB=1,IA
      DO 33 IC=1,IA
      DO 32 ID=1,IB
   32 A(IC,ID)=FX(IC,ID)
      DO 33 IE=IB,IA
   33 A(IC,IE)=FX(IC,IE+1)
```

```
      CALL MATINV(A,IA,WX,0,DETERM)
 34   F(IB)=DETERM
      DO 35 IB=1,IA
      DO 35 IC=1,IA
 35   A(IB,IC)=FX(IB,IC)
      CALL MATINV(A,IA,WX,0,DETERM)
      F(JIGEN)=DETERM
      SUM=0.0
      DO 36 IB=1,JIGEN
 36   SUM=SUM+F(IB)**2
      SING=SQRT(SUM)
      GO TO 38
 37   SING=SQRT(FX(1,1)**2+FX(1,2)**2)
 38   IF(SING-CONST) 39,40,40
 39   CALL KANSU(X,F,M)
      WRITE(6,2000)  SING,(X(IA),F(IA),IA=1,M)
      GO TO 123
 40   CALL KUTTA(X,WX,EH,KR,JIGEN,2,LYESNO,YL,XM(2),1)
      JIG=JIGEN
      DO 41 IA=1,JIG
 41   RUNGE(IA,1)=WX(IA)
      DO 42 IB=1,JIG
 42   RUNGX(IB)=X(IB)+RUNGE(IB,1)/2.0
      CALL KUTTA(RUNGX,WX,EH,KR,JIG,2,LYESNO,YL,XM(2),1)
      DO 43 IC=1,JIG
 43   RUNGE(IC,2)=WX(IC)
      DO 44 ID=1,JIG
 44   RUNGX(ID)=X(ID)+RUNGE(ID,2)/2.0
      CALL KUTTA(RUNGX,WX,EH,KR,JIG,2,LYESNO,YL,XM(2),1)
      DO 45 IE=1,JIG
 45   RUNGE(IE,3)=WX(IE)


      DO 46 IA=1,JIG
 46   RUNGX(IA)=X(IA)+RUNGE(IA,3)
      CALL KUTTA(RUNGX,WX,EH,KR,JIG,2,LYESNO,YL,XM(2),1)
      DO 47 IB=1,JIG
 47   RUNGE(IB,4)=WX(IB)
      DO 48 IC=1,JIG
 48   WX(IC)=X(IC)+(RUNGE(IC,1)+2.0*(RUNGE(IC,2)+RUNGE(IC,3))+
     +        RUNGE(IC,4))/6.0
C     STEP-SIZE CONTROL
      EM=XH(JIG)*0.25
      IF(ABS(WX(JIG)-X(JIG))-EM) 50,49,49
 49   EH=EH*0.5
      GO TO 30
 50   CALL KANSU(X,F,JIGEN)
 51   JIH=JIGEN-1
      EM=0.1E-05
      DO 52 IC=1,JIH
 52   EM=AMAX1(EM,ABS(F(IC)))
      IF(EM-0.1E-03 ) 54,53,53
 53   CALL NEWTON(JIH,X,KONV,CONST)
      IF(KONV) 123,123,30
 54   CALL KANSU(WX,F,JIGEN)
      DO 55 IC=1,JIH
 55   EN=AMAX1(0.1E-07 ,ABS(F(IC)))
      IF(EN-0.1E-02 ) 56,56,49
 56   GG2(2)=F(JIG)
      IF(GG2(1)*GG2(2)) 96,96,57
 57   DEX=WX(JIG)-XIR(JIG)
C     RIGHT BOUNDARY TEST
      IF(DEX) 66,66,58
 58   GOSA=EM*ABS(WX(JIG))
      IF(GOSA-EPSIL ) 59,59,60
 59   GOSA=EPSIL
 60   CONTINUE
```

```
      IF(DEX-GOSA) 64,64,61
   61 IF(JIG-2) 63,63,62
   62 CALL NEWTON(JIH,X,KONV,CONST)
      IF(KONV) 123,123,63
   63 IF(EH-EPSIL) 64,64,98
   64 JIH=JIGEN
C     DELIVER THE DATA
      WALPHA=AMAX1(WALPHA,EM)
      KKS(JIH)=KKS(JIH)+1
      JB=KKS(JIH)
      DO 65 IB=1,M
   65 YYF(IB,JB)=WX(IB)
      KKR(JB)=KR
      GO TO 123
   66 JIH=JIGEN
C     CHECK WHETHER THE INTEGRAL CURVE RETURNS THE START LINE
      SXIL=(WX(JIH)-XIL(JIH))*(X(JIH)-XIL(JIH))
      IF(SXIL) 70,67,92
   67 IF(JIH-2) 69,69,68
   68 IF(WX(JIH)-XIL(JIH)) 92,80,92


   69 IF(WX(2)-XIL(2)) 92,74,92
   70 GOSA=EM*ABS(X(JIH))
      IF(GOSA-EPSIL) 71,71,72
   71 GOSA=EPSIL
   72 CONTINUE
      IF(ABS(WX(JIH)-X(JIH))-GOSA) 74,74,73
   73 IF(EH-EPSIL) 74,74,98
   74 IF(KS(JIH)) 86,86,75
   75 JB=KS(JIH)
      DO 79 LM=1,JB
      DO 78 IC=1,JIH
      GOSA=ALPHA*ABS(WX(IC))
      IF(GOSA-EPSIL) 76,76,77
   76 GOSA=EPSIL
   77 CONTINUE
      IF(ABS(YF(IC,LM)-WX(IC))-GOSA) 78,78,79
   78 CONTINUE
      GO TO 89
   79 CONTINUE
      GO TO 86
   80 IF(IANS(1)) 123,123,81
   81 JB=IANS(1)
      DO 84 LM=1,JB
      DO 84 IC=1,JIH
      GOSA=ALPHA*ABS(WX(IC))
      IF(GOSA-EPSIL) 82,82,83
   82 GOSA=EPSIL
   83 CONTINUE
      IF(ABS(ANSX(IC,LM,1)-WX(IC))-GOSA) 85,85,84
   84 CONTINUE
      GO TO 123
   85 KRANS(LM,1)=3
      GO TO 123
   86 JIS=JIGEN
      KS(JIS)=KS(JIS)+1
      JB=KS(JIS)
      IF(JB-MEMORY) 866,866,112
  866 DO 87 IB=1,M
   87 YF(IB,JB)=WX(IB)
      KSR(JB)=KR
      IF(JIGEN-2) 123,123,88
   88 H=DELTAS
      EH=H
      GO TO 92
   89 IF(KSR(LM)) 90,91,90
```

```
   90    KRANS(LM,1)=3
         KSR(LM)=3
         GO TO 123
   91    KRANS(LM,1)=KR
         KSR(LM)=KR
         GO TO 123
   92    GG2(1)=GG2(2)
         JIS=JIGEN
         DO 93 IB=1,JIS


   93    X(IB)=WX(IB)
C        TEST FOR BOUNDARY
         DO 94 IC=1,JIS
         IF(X(IC)-DXM(IC)) 94,94,123
   94    CONTINUE
         DO 95 IC=1.JIS
         IF(DXL(IC)-X(IC)) 95,95,123
   95    CONTINUE
         IF( X(JIS)-XIL(JIS)+XH(JIS) ) 123,123,30
   96    EN=ABS(GG2(1))
         IF(EN-EM) 99,97,97
   97    IF(EH-EPSIL) 99,99,98
   98    EH=EH*0.125
         GO TO 30
   99    JIK=JIGEN
C        NEWTON METHOD
         CALL NEWTON(JIGEN,X,KONV,CONST)
         IF(KONV) 117,101,101
  101    CALL KANSU(X,F,M)
         IF(JIK-M) 103,102,102
  102    WRITE(6,3000) JIGEN,(X(LM),LM=1,M),(F(LM),LM=1,M)
  103    EN=(X(JIK)-XIL(JIK)+XH(JIK))*(X(JIK)-XIR(JIK))
         IF(EN) 105,105,104
  104    WRITE(6,4000)
         GO TO 118
  105    JB=IANS(JIK)
         IF(JB) 106,111,106
  106    DO 110 IC=1,JB
         DO 109 LM=1,M
         GOSA=WALPHA*ABS(X(LM))
         IF(GOSA-EPSIL) 107,107,108
  107    GOSA=EPSIL
  108    CONTINUE
         IF(ABS(ANSX(LM,IC,JIK)-X(LM))-GOSA) 109,109,110
  109    CONTINUE
         GO TO 121
  110    CONTINUE
  111    IANS(JIK)=IANS(JIK)+1
         JB=IANS(JIK)
         IF(JB-MEMORY)  113,113,112
  112    WRITE(6,5000) JB
         NUM=-1
         GO TO 9000
  113    DO 114 LM=1,M
  114    ANSX(LM,JB,JIK)=X(LM)
         KRANS(JB,JIK)=0
  115    DO 116 LN=1,M
  116    WX(LN)=X(LN)
         GO TO 121
  117    WRITE(6,5100)
  118    CALL KANSU(WX,F,M)
         WRITE(6,3000) JIGEN,(WX(LM),LM=1,M),(F(LM),LM=1,M)
         WRITE(6,5200)
         IF(ABS(F(JIGEN))-0.1E-03 ) 119,119,123
```

```
 119    IANS(JIK)=IANS(JIK)+1
        JB=IANS(JIK)
        IF(JB-MEMORY)  1199,1199,112
1199    KRANS(JB,JIK)=0
        DO 120 LM=1,M
 120    ANSX(LM,JB,JIK)=WX(LM)
 121    GG2(1)=GG2(2)
        DO 122 LN=1,M
 122    X(LN)=WX(LN)
        GO TO 29
 123    IF(KKRR) 124,100;124
 100    CONTINUE
 124    IF(JIGEN-2) 125,125,204
 125    IF(XM(1)-YL) 127,127,126
 126    CONTINUE
        GO TO 12
 127    JIK=JIGEN
        IF(KKS(JIK)) 128,131,128
 128    JB=KKS(JIK)
        DO 130 LMN=1,JB
        DO 129 IC=1,M
 129    YF(IC,LMN)=YYF(IC,LMN)
 130    KSR(LMN)=KKR(LMN)
        KS(JIK)=KKS(JIK)
        ALPHA=WALPHA
        IF(JIK-M) 200,132,132
 131    KS(JIK)=0
        ALPHA=0.1E-04
        IF(JIK-M) 200,132,132
 132    WRITE(6,6000) LLJ(JIK),JIK
 200    CONTINUE
        IF(M-2)  501,501,201
 201    JIK=JIGEN
 202    IF(JIK-2) 215,215,204
 203    JIM=JIL-1
        IF(IANS(JIM)) 205,222,205
 204    IANS(1)=IANS(1)-1
        IF (IANS(1))216,216,208
 205    IANS(1)=IANS(JIM)
        IANS(JIM)=0
        JB=IANS(1)
        KS(JIGEN)=IANS(1)
        DO 207 IA=1,JB
        DO 206 IB=1,M
        YF(IB,IA)=ANSX(IB,IA,JIM)
 206    ANSX(IB,IA,1)=ANSX(IB,IA,JIM)
        KSR(IA)=KRANS(IA,JIM)
 207    KRANS(IA,1)=KRANS(IA,JIM)
 208    JB=IANS(1)
        DO 209 IA=1,M
 209    X(IA)=ANSX(IA,JB,1)
        IF(KRANS(JB,1)-2) 210 ,213 ,204
 210    IF(KRANS(JB,1)-1) 211,212,212
 211    KKRR=0


        GO TO 214
 212    KKRR=1
        GO TO 214
 213    KKRR=2
 214    CONTINUE
        GO TO 752
 215    JIGEN=3
        JIL=3
 216    IF(KKS(JIL)) 218,217,218
 217    ALPHA=0.1E-04
        KS(JIL)=0
```

```
      GO TO 203
218   JIM=JIL-1
      IANS(JIM)=KKS(JIL)
      JB=IANS(JIM)
      DO 220 IA=1,JB
      DO 219 IB=1,M
219   ANSX(IB,IA,JIM)=YYF(IB,IA)
220   KRANS(IA,JIM)=KKR(IA)
221   ALPHA=WALPHA
222   IF(JIGEN-M) 224,223,223
223   WRITE(6,6000) LLJ(JIGEN),JIGEN
224   IF(JIGEN-4) 225,302,403
225   JIGEN=2
300   CONTINUE
      IF(M-3)  501,501,226
226   JIGEN=4
      JIL=4
      IF(KKS(4)) 401,301,401
301   ALPHA=0.1E-04
      KS(4)=0
      M402=0
      GO TO 755
302   JIGEN=2
400   CONTINUE
      IF(M-4)  501,501,227
227   JIGEN=5
      JIL=5
      IF(KKS(5)) 401,402,401
401   M402=1
      GO TO 757
402   ALPHA=0.1E-03
      M402=0
      KS(5)=0
      GO TO 757
403   JIGEN=2
500   CONTINUE
501   WRITE(6,7000) MCUT
      JB=IANS(M)
      IF(JB) 502,505,502
502   DO 504 K=1,JB
      DO 503 I=1,M
503   X(I)=ANSX(I,K,M)
      CALL KANSU(X,F,M)


504   WRITE(6,3000) M,(X(LM),LM=1,M),(F(LM),LM=1,M)
      GO TO 506
505   WRITE(6,7100)
506   MANS(MCUT)=IANS(M)
      IF(MCUT-1)  507,508,507
507   I=MCUT-1
      IF(MANS(MCUT)-MANS(I))  508,510,508
508   DO 509 I=1,M
509   XH(I)=XH(I)*0.5
999   CONTINUE
      NUM=IANS(M)
      GO TO 9000
510   WRITE(6,8000)
      NUM=IANS(M)
1200  FORMAT(1H1,3X,12HDATA(REGION),(/3F20.10))
1300  FORMAT(1H0,3X,14HDATA(BOUNDARY),(/2F20.10))
1400  FORMAT(1H0,//,3X,6HEPSIL=,E20.10,3X,6HCONST=,E20.10)
2000  FORMAT(1H ,20X,25HTHERE IS A SINGULAR POINT /20X,5HSING=,E20.10
     +       //20X,1HX,(/15X,2E20.10))
3000  FORMAT(1H ,3X,6HJIGEN=,I3,(/5X,5E20.10))
4000  FORMAT(1H ,30HABOVE IS OUT OF DOMAIN SECTION)
5000  FORMAT(1H ,5X,I5,3X,10HNO EFFECTS)
```

```
 5100     FORMAT(1H ,20X,17HNON   CONVERGENCE)
 5200     FORMAT(1H ,23HABOVE IS STARTING POINT)
 6000     FORMAT(1H ,20X,16HABOVE IS SECTION,I4,1H(,I4,1H))
 7000     FORMAT(1H0,/, 5X25H*****ANSWER  FOR  KIZAMI=,I3)
 7100     FORMAT(1H ,18H*****NOT  OBTAINED)
 8000     FORMAT(1H0,20X,15HABOVE IS ANSWER)
 9000     RETURN
          END
C         SUBPROGRAM OF ONE-DIMENSION CASE AND RUNGE-KUTTA METHOD
C         AS JCOUNT=0, THIS SUBPROGRAM COMPUTES A ROOT OF THE SINGLE
C         EQUATION F(X)=0  ON THE INTERVAL (YL,YM).
C
C         AS JCOUNT=1, THIS SUBPROGRAM COMPUTES THE VALUE OF FUNCTION
C         WHICH IS USED IN THE RUNGE-KUTTA METHOD.
C
C         DIMENSION   X(MAX),F(MAX),FX(MAX,MAX),A(MAX,MAX),WX(MAX),G(2)
C         COMMON  EPSIL,F,FX
C

          SUBROUTINE KUTTA(X,WX,EH,KR,JIGEN,J,LYESNO,YL,YM,JCOUNT)
          DIMENSION X(5),F(5),FX(5,5),A(5,5),WX(5),G(2)
          COMMON  EPSIL,F,FX
          IF( JCOUNT-1) 100,200,200
  100     EPS=0,1E-04
          DO 40 IB=1,3
          DELTAG=0.03125
          X(J)=YL
          CALL KANSU(X,F,1)
          G(1)=F(1)
          DO 30 IC=1,8
    1     X(J)=X(J)+DELTAG
          CALL KANSU(X,F,1)
          G(2)=F(1)
          IF(G(1)*G(2) ) 3,3,2
    2     G(1)=G(2)
          IF(YM-X(J) ) 6,1,1
    3     IF(ABS(G(1))-EPS) 5,4,4
    4     GOSA=ABS(X(J))*EPSIL
          IF(GOSA-EPSIL) 23,23,24
   23     GOSA=EPSIL
   24     IF(ABS(DELTAG)-GOSA) 5,5,25
   25     X(J)=X(J)-DELTAG
          DELTAG=DELTAG*0,125
          IF(ABS(DELTAG)-GOSA) 5,30,30
   30     CONTINUE
          EPS=EPS*10.0
   40     CONTINUE
          GO TO 6
    5     LYESNO=1
          GO TO 7
    6     LYESNO=-1
    7     RETURN
  200     IA=JIGEN-1
          CALL BIBUN(X,FX,IA)
          DO 13 IB=1,IA
          DO 12 IC=1,IA
          DO 10 ID=1,IB
   10     A(IC,ID)=FX(IC,ID)
          DO 11 IE=IB,IA
   11     A(IC,IE)=FX(IC,IE+1)
   12     CONTINUE
          IF( IA-1) 20,20,21
   20     F(1)=FX(1,2)
          F(2)=FX(1,1)
          GO TO 22
   21     CALL MATINV(A,IA,WX,0,DETERM)
   13     F(IB)=DETERM
          DO 15 IB=1,IA
```

```
      DO 14 IC=1,IA
   14 A(IB,IC)=FX(IB,IC)
   15 CONTINUE
      CALL MATINV(A,IA,WX,0,DETERM)
      F(JIGEN)=DETERM



   22 SUM=0,0
      DO 16 IB=1,JIGEN
   16 SUM=SUM+F(IB)**2
      SUM=SQRT(SUM)
      IF( KR-2) 18,17,17
   17 SUM=-SUM
   18 DO 19 IE=1,JIGEN
   19 WX(IE)=F(IE)*EH/SUM*(-1,0)**IE
      RETURN
     END
C       NEWTON METHOD
        SUBROUTINE NEWTON(JIGEN,X,KONV,CONST)
C
C       DIMENSION  X(MAX),F(MAX),FX(MAX,MAX)
C       COMMON  EPSIL,F,FX
C
      DIMENSION X(5),F(5),FX(5,5)
      COMMON  EPSIL,F,FX
      EPS=CONST
      DO 20 KCOUNT=1,2
      DO 15 ITERA=1,20
      CALL KANSU(X,F,JIGEN)
      CALL BIBUN(X,FX,JIGEN)
      DO 10 I=1,JIGEN
   10 F(I)=-F(I)
      CALL MATINV(FX,JIGEN,F,1,DETERM)
      DO 11 J=1,JIGEN
      GOSA=EPS*ABS(X(J))
      IF(GOSA-EPSIL)  16,17,17
   16 GOSA=EPSIL
   17 CONTINUE
      IF( ABS(F(J))-GOSA) 11,11,12
   11 CONTINUE
      GO TO 14
   12 DO 13 J=1,JIGEN
   13 X(J)=X(J)+F(J)
   15 CONTINUE
      EPS=EPS*10,0
   20 CONTINUE
      KONV=-1
      RETURN
   14 KONV=1
      RETURN
     END
C       MATRIX COMPUTATION
C       MATINV(A,N,B,M,DETERM)
C       GAUSS-JORDAN METHOD.
C       AS M=0, THE DETERMINANT OF MATRIX A OF ORDER N IS PLACED IN
C       DETERM AND NO SIMULTANEOUS SOLUTIONS ARE CALLED FOR.
C       AS M=1, THE VECTOR B CONTAINS THE CONSTANT VECTOR WHEN MATINV IS
C       CALLED , AND THIS IS REPLACED WITH SOLUTION VECTOR .
C
C       DIMENSION  IPIVOT(MAX),A(MAX,MAX),B(MAX),INDEX(MAX,2),PIVOT(MAX)
C       COMMON EPSIL
C
        SUBROUTINE MATINV(A,N,B,M,DETERM)
```

```
        DIMENSION IPIVOT(5 ),A(5,5),B(5),INDEX(5 ,2),PIVOT(5 )
        COMMON      EPSIL
C       INITIALIZATION
        DETERM=1.0
        DO 20 J=1,N
   20   IPIVOT(J)=0
        DO  37 I=1,N
C       SEARCH FOR PIVOT ELEMENT
        AMAX=0.0
        DO  25 J=1,N
        IF( IPIVOT(J)-1) 21,25,21
   21   DO  24 K=1,N
        IF( IPIVOT(K)-1) 22,24,40
   22   IF( ABS(AMAX)-ABS(A(J,K))) 23,24,24
   23   IROW=J
        ICOLUM=K
        AMAX=A(J,K)
   24   CONTINUE
   25   CONTINUE
        IF(ABS(AMAX)-EPSIL) 38,38,255
  255   IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C       INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
        IF( IROW-ICOLUM) 26,29,26
   26   DETERM=-DETERM
        DO 27  L=1,N
        SWAP=A(IROW,L)
        A(IROW,L)=A(ICOLUM,L)
   27   A(ICOLUM,L)=SWAP
        IF(M) 28,29,28
   28   SWAP=B(IROW)
        B(IROW)=B(ICOLUM)
        B(ICOLUM)=SWAP
   29   INDEX(I,1)=IROW
        INDEX(I,2)=ICOLUM
        PIVOT(I)=A(ICOLUM,ICOLUM)
        DETERM=DETERM*PIVOT(I)
C       DIVIDE PIVOT ROW BY PIVOT ELEMENT
        DO 30 L=1,N
   30   A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)
        IF(M) 31,32,31
   31   B(ICOLUM)=B(ICOLUM)/PIVOT(I)
C       REDUCE NON PIVOT ROWS
   32   DO 36  L1=1,N
        IF( L1-ICOLUM) 33,36,33
   33   T=A(L1,ICOLUM)
        DO  34 L=1,N
   34   A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
        IF( M ) 35,36,35
   35   B(L1)=B(L1)-B(ICOLUM)*T
   36   CONTINUE
   37   CONTINUE
        GO TO 41
   38   IF(M) 40,39,40
   39   DETERM=0.0
        GO TO 41
   40   WRITE(6,1000)
 1000   FORMAT(1H ,15H SINGULAR  CASE)
   41   RETURN
        END
C       SUBPROGRAM OF GIVEN EQUATIONS
```

```
      SUBROUTINE KANSU(X,F,KOSU)
      DIMENSION X(5),F(5)
      P=X(1)
      Q=X(2)
      R=X(3)
      S=X(4)
      T=X(5)
      F(1)=P**3-2.0*P*Q+0.75*P+R+1.0
      IF(KOSU-2) 20,12,12
   12 F(2)=Q*P**2-Q**2+0.75*Q-P*R+S+0.25
      IF(KOSU-3) 20,13,13
   13 F(3)=R*P**2-Q*R+0.75*R-P*S+T+0.75
      IF(KOSU-4) 20,14,14
   14 F(4)=S*P**2-Q*S+0.75*S-P*T
      IF(KOSU-5) 20,15,15
   15 F(5)=T*P**2-Q*T+0.75*T-0.25
   20 RETURN
      END
C     SUBROUTINE OF DERIVATIVES
      SUBROUTINE BIBUN(X,FX,KOSU)
      DIMENSION X(5),FX(5,5)
      P=X(1)
      Q=X(2)
      R=X(3)
      S=X(4)
      T=X(5)
      FX(1,1)=3.0*P**2-2.0*Q+0.75
      FX(1,2)=-2.0*P
      FX(1,3)=1.0
      FX(1,4)=0.0
      FX(1,5)=0.0
      IF(KOSU-2) 20,12,12
   12 FX(2,1)=2.0*P*Q-R
      FX(2,2)=P**2-2.0*Q+0.75
      FX(2,3)=-P
      FX(2,4)=1.0
      FX(2,5)=0.0
      IF(KOSU-3) 20,13,13
   13 FX(3,1)=2.0*P*R-S
      FX(3,2)=-R
      FX(3,3)=P**2-Q+0.75
      FX(3,4)=-P
      FX(3,5)=1.0
      IF(KOSU-4) 20,14,14
   14 FX(4,1)=2.0*P*S-T
      FX(4,2)=-S
      FX(4,3)=0.0
      FX(4,4)=P**2-Q+0.75
      FX(4,5)=-P
      IF(KOSU-5) 20,15,15
   15 FX(5,1)=2.0*P*T
      FX(5,2)=-T
      FX(5,3)=0.0
      FX(5,4)=0.0
      FX(5,5)=P**2-Q+0.75
   20 RETURN
      END
```