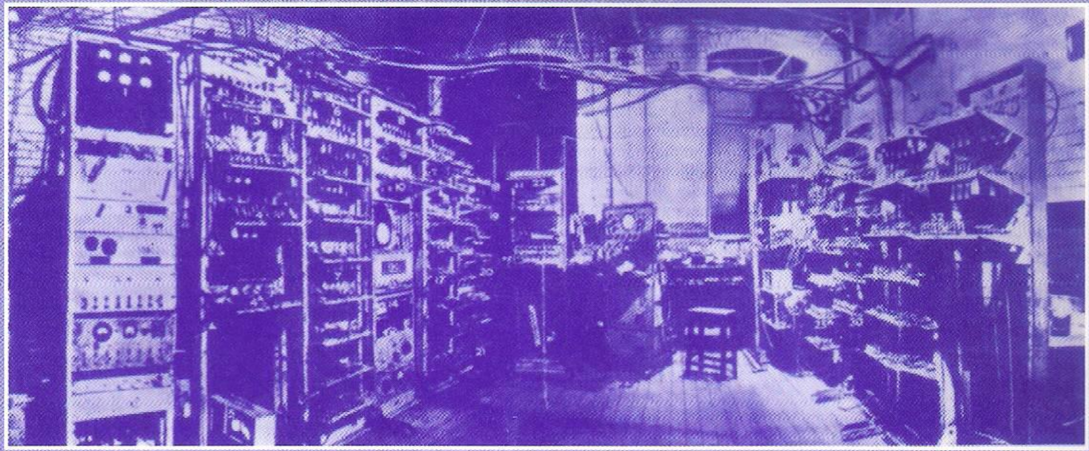
PROGRAMMING IN FORTRAN 90

A First Course for Engineers and Scientists



I. M. Smith

Programming in Fortran 90

A First Course for Engineers and Scientists

I. M. Smith
University of Manchester, UK

JOHN WILEY & SONS
Chichester · New York · Brisbane · Toronto · Singapore

Reprinted March 1995

Cover illustration The enhanced Mark 1 in June 1949 at the University of Manchester (© National Computing Centre, Manchester)

Other Wiley Editorial Offices

John Wiley & Sons, Inc., 605 Third Avenue, New York, NY 10158-0012, USA

Jacaranda Wiley Ltd, 33 Park Road, Milton, Queensland 4064, Australia

John Wiley & Sons (Canada) Ltd, 22 Worcester Road, Rexdale, Ontario M9W 1L1, Canada

John Wiley & Sons (SEA) Pte Ltd, 37 Jalan Pemimipin #05-04, Block B, Union Industrial Building, Singapore 2057

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

ISBN 0 471 94185 9

Typeset in 10/12 pt times by Thomson Press (India) Ltd, New Delhi
Printed and bound in Great Britain by Redwood Books, Trowbridge, Wiltshire.

se study 7: eigenvalues

sider vibration of three railway wagons as shown. For a force f_n applied to on n, we have static displacements given by

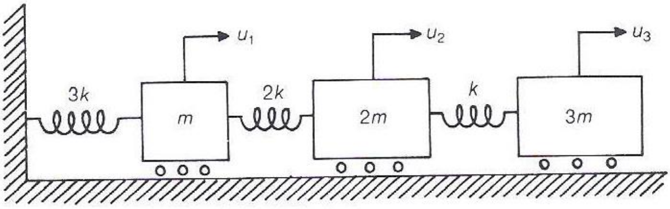


Figure. Vibrational system with three degrees of freedom

$$u_1 = \frac{1}{3k} f_1 + \frac{1}{3k} f_2 + \frac{1}{3k} f_3$$

$$u_2 = \frac{1}{3k} f_1 + \frac{5}{6k} f_2 + \frac{5}{6k} f_3$$

$$u_3 = \frac{1}{3k} f_1 + \frac{5}{6k} f_2 + \frac{11}{6k} f_3$$

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} \frac{1}{3k} & \frac{1}{3k} & \frac{1}{3k} \\ \frac{1}{3k} & \frac{5}{6k} & \frac{5}{6k} \\ \frac{1}{3k} & \frac{5}{6k} & \frac{11}{6k} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

$u = Mf$ where M is the "influence matrix" and its elements are "influence efficients".

f no external forces are applied, we have $f_n = -m_n \ddot{u}$ (where $m_1 = m, m_2 = 2m,$ $= 3m$).

For a natural vibration,

$$u_n = x_n \sin(\omega t + \phi)$$
$$\therefore \ddot{u}_n = -\omega^2 x_n \sin(\omega t + \phi)$$
$$\therefore f_n = m_n \omega^2 x_n \sin(\omega t + \phi)$$

$$\therefore \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{1}{3k} & \frac{1}{3k} & \frac{1}{3k} \\ \frac{1}{3k} & \frac{5}{6k} & \frac{5}{6k} \\ \frac{1}{3k} & \frac{5}{6k} & \frac{11}{6k} \end{pmatrix} \begin{pmatrix} m\omega^2 x_1 \\ 2m\omega^2 x_2 \\ 3m\omega^2 x_3 \end{pmatrix} = \frac{m\omega^2}{k} \begin{pmatrix} \frac{1}{3} & \frac{2}{3} & 1 \\ \frac{1}{3} & \frac{5}{3} & \frac{5}{2} \\ \frac{1}{3} & \frac{5}{3} & \frac{11}{2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

or $X = \lambda GX$ where $\lambda = m\omega^2/k$ and $1/\lambda$ is an eigenvalue of G (or λ is an eigenvalue G^{-1}).

We wish to find the lowest natural frequency of the system. In this mode, we expe wagon 3 to have the greatest displacement. Therefore, let us try

$$X = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

If we now compute $X^* = GX$, and scale X^* so that its third element is 1, we m check whether X^*, as scaled, is the same as X(or, in practice, sufficiently close). If X is an eigenvector and the scaling factor is the eigenvalue $1/\lambda$. Otherwise, we again using X^* in place of X. It can be shown that the process converges on eigenvalue $1/\lambda$ of largest absolute value, corresponding to the lowest value of ω, required.

Write a program to find the lowest natural frequency $= (\omega/2\pi)$ of a system n railway wagons, given their masses $(m_1, m_2, \ldots, m_n)$, the stiffness of their spri $(k_1, k_2, \ldots, k_n)$, and an initial approximation to the eigenvector. Test your proced on the three wagon problem above.

Case study 8: Runge-Kutta

A popular method for solving systems of (non-linear) ordinary differential equati is the "Runge-Kutta" method of the fourth order (see for example Griffiths Smith, 1991, p 226*). In general, such a system will be of the form

$$\frac{dy_i}{dx} = f_i(x, y_0, y_1, \ldots, y_{n-1}) \qquad i = 0, 1, 2, \ldots, n$$

with n initial conditions $y_i(x_0) = A_i$, $i = 0, 1, 2, \ldots, n$.

For example the system of two equations:

$$\frac{dy}{dx} = f(x, y, z) \qquad y(x_0) = y_0$$

$$\frac{dz}{dx} = g(x, y, z) \qquad z(x_0) = z_0$$

may be solved by advancing the solution of y and z to $x_1 = x_0 + h$ by the form

$$y(x_1) = y(x_0) + K$$
$$z(x_1) = z(x_0) + L$$

* Griffiths, D.V. and Smith, I.M. *Numerical Methods for Engineers*, Blackwell, 1991

re $K = (K_0 + 2K_1 + 2K_2 + K_3)/6$ and $L = (L_0 + 2L_1 + 2L_2 + L_3)/6$ and

$$K_0 = hf(x_0, y_0, z_0)$$
$$L_0 = hg(x_0, y_0, z_0)$$
$$K_1 = hf(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}K_0, z_0 + \tfrac{1}{2}L_0)$$
$$L_1 = hg(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}K_0, z_0 + \tfrac{1}{2}L_0)$$
$$K_2 = hf(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}K_1, z_0 + \tfrac{1}{2}L_1)$$
$$L_2 = hg(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}K_1, z_0 + \tfrac{1}{2}L_1)$$
$$K_3 = hf(x_0 + h, y_0 + K_2, z_0 + L_2)$$
$$L_3 = hg(x_0 + h, y_0 + K_0, z_0 + L_2)$$

te a program to solve the pair of equations

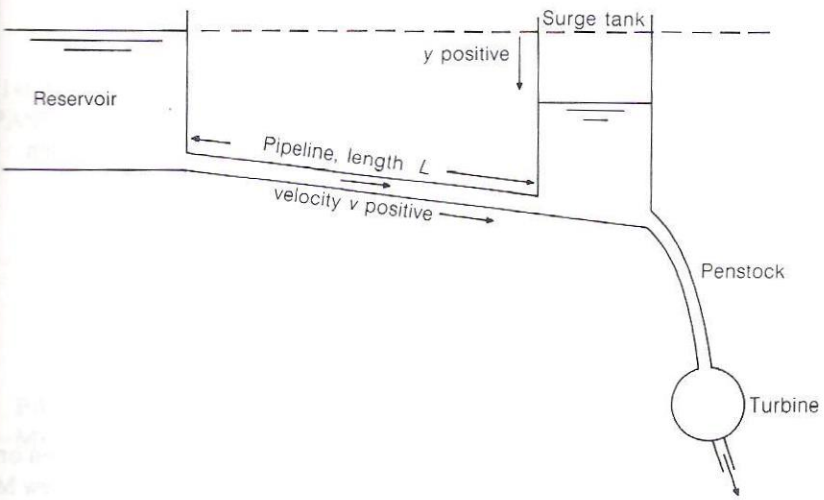
$$\frac{dy}{dx} = 3xz + 4, \qquad y(0) = 4$$

$$\frac{dz}{dx} = xy - z - e^x, \qquad z(0) = 1$$

this method at $x = 0.5$, using steps of 0.1.

se study 9: water levels in a turbine

"surge tank" is a device for damping out the effects of suddenly starting or pping a turbine in the arrangement shown below:

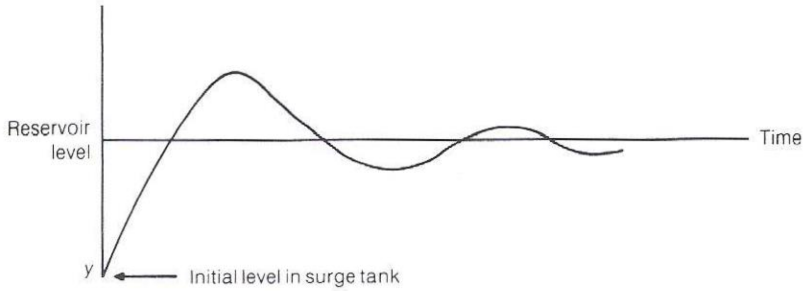


ith normal flow through the turbine, the level in the surge tank is below the servoir level by the amount of friction head loss h_f in the pipeline, so that initially $= h_f = 4fLv^2/2gd$. If the turbine is suddenly shut down the flow in the penstock is stopped and the momentum of the water in the pipeline carries it into the surge tan causing the water level to rise. A pressure gradient builds up against the pipeline flo and decelerates it. The motions of the water in the pipeline and the water level in th surge tank are described by the equations:

$$A\frac{\mathrm{d}y}{\mathrm{d}t} = -va + X_t$$

$$\frac{L}{g}\frac{\mathrm{d}v}{\mathrm{d}t} = y - h_f$$

where X_t is the rate of flow through the turbine (zero after shut-down). Typically th level in the surge tank will oscillate as shown below:



Given that the reservoir level remains constant, and that

$$L = 1200\,\text{m}$$
$$d = 800\,\text{mm}$$
$$D = 2.5\,\text{m}$$
$$f = 0.005$$
$$X_t\ \text{(before shut-down)} = 1\,\text{m}^3/\text{sec}$$

use the Runge-Kutta method of case study 8 to calculate what will happen

(a) when the turbine is shut down after steady running
(b) when the turbine is started after long quiescence.

Hence design a suitable height of surge tank.

Case study 10: date of Easter

Easter Day in any given year is the Sunday following the first Full Moon on or af March 21st. A Full Moon is deemed to occur 13 days after the preceding New Mo The date of the last New Moon in January may be calculated from the "Epact" f the year, and from that the new moons in February, March and April may obtained. The exact method is given in the Structure Chart below. All dates are h as the number of days from the beginning of the year.

Printed and bound in Great Britain by Redwood Books, Trowbridge, Wiltshire.

```
    END DO
    A(I,J)=A(I,J)-SUM
  END DO
END DO
! forward substitution stage
! this depends on the assumption l(i,i)=1.0
  DO I=1,N
    X(I)=B(I)
    B(I+1:N)=B(I+1:N)-A(I+1:N,I)*X(I)
  END DO
! backward substitution stage
  DO I=N,1,-1
    X(I)=B(I)/A(I,I)
    B(N-1:1:-1)=B(N-1:1:-1)-A(N-1:1:-1,I)*X(I)
  END DO
END SUBROUTINE ELIMINATE
END PROGRAM COMPLEX_GAUSSIAN_ELIMINATION
```

The solution is:

$$\begin{bmatrix} (\frac{3}{2}, & -\frac{1}{2}) \\ (1.0, & 1.0) \\ (-\frac{1}{2}, & -\frac{5}{2}) \end{bmatrix}$$

7.
```
PROGRAM EIGENVALUES
! compute natural frequencies by a simple iterative method
IMPLICIT NONE
REAL,ALLOCATABLE::A(:,:),X0(:),X1(:)
INTEGER::,N,ITERS,ITS;LOGICAL::CONVERGED
READ::TOL,BIG
READ*,N,TOL,ITS
ALLOCATE(A(N,N),X0(N),X1(N))
READ*,A;READ*,X0
ITERS=0
  DO
    ITERS=ITERS+1
    X1=MATMUL(A,X0)
    BIG=MAXVAL(X1);IF(ABS(MINVAL(X1))>BIG)BIG=MINVAL(X1)
          X1=X1/BIG
    CONVERGED=(MAXVAL(ABS(X1-X0))/MAXVAL(ABS(X1))<TOL)
    X0=X1
    IF(CONVERGED.OR.ITERS==ITS)EXIT
  END DO
X1=X1/SQRT(SUM(X1**2))
PRINT*,BIG;PRINT*,X1;PRINT*,ITERS
END PROGRAM EIGENVALUES
```

For a "tolerance" TOL of 10^{-6}, λ converges to 6.4673 in 8 iterations.

8.
```
PROGRAM RUNGE_KUTTA
! 4th order method for systems of equations
IMPLICIT NONE
REAL,ALLOCATABLE::Y(:),Y0(:),K0(:),K1(:),K2(:),K3(:)
INTEGER::N,STEPS,I,J
REAL::H,X
READ*,N,STEPS,H,X
ALLOCATE(Y(N),Y0(N),K0(N),K1(N),K2(N),K3(N))
READ*,Y
PRINT*,'*****************SYSTEMS OF EQUATIONS*****************'
PRINT*,'***********4TH ORDER RUNGE-KUTTA METHOD***********'
PRINT*,'  X  Y(I)  ,I=1,',N
DO J=0,STEPS
  PRINT'(5E13.5)',X,(Y(I),I=1,N)
  K0=FUNC(X,Y,N);Y0=Y;Y=Y0+.5*H*K0;X=X+.5*H
  K1=FUNC(X,Y,N);Y=Y0+.5*H*K1
  K2=FUNC(X,Y,N);Y=Y0+H*K2;X=X+.5*H
  K3=FUNC(X,Y,N);Y=Y0+(K0+2.*(K1+K2)+K3)/6.*H
END DO
CONTAINS
FUNCTION FUNC(X,Y,N)
! provides the values of f(x,y(i)) specified by the user
IMPLICIT NONE
INTEGER,INTENT(IN)::N
REAL::FUNC(N)
REAL,INTENT(IN)::X,Y(:)
FUNC(1)=3.0*X*Y(2)+4.0
FUNC(2)=X*Y(1)-Y(2)-EXP(X)
RETURN
END FUNCTION FUNC
END PROGRAM RUNGE_KUTTA
```

The results are $y(0.5)=6.2494$, $z(0.5)=0.67386$.

9.
```
PROGRAM SURGE_TANK
! oscillations in a tank by a 4th order method for systems of
equations
IMPLICIT NONE
REAL,ALLOCATABLE::Y(:),Y0(:),K0(:),K1(:),K2(:),K3(:)
INTEGER::N,STEPS,I,J
REAL::H,X
READ*,N,STEPS,H,X
ALLOCATE(Y(N),Y0(N),K0(N),K1(N),K2(N),K3(N))
READ*,Y
PRINT*,'*****************SYSTEMS OF EQUATIONS*****************'
PRINT*,'***********4TH ORDER RUNGE-KUTTA METHOD***********'
PRINT*,'    x    Y(I)   ,I=1,',N
DO J=0,STEPS
  PRINT'(5E13.5)',X,(Y(I),I=1,N)
  K0=FUNC(X,Y,N);Y0=Y;Y=Y0+.5*H*K0;X=X+.5*H
  K1=FUNC(X,Y,N);Y=Y0+.5*H*K1
```

```
  K2=FUNC(X,Y,N);Y=Y0+H*K2;X=X+.5*H
  K3=FUNC(X,Y,N);Y=Y0+(K0+2.*(K1+K2)+K3)/6.*H
END DO
CONTAINS
FUNCTION FUNC(X,Y,N)
! provides the values of f(x,y(i)) specified by the user
IMPLICIT NONE
INTEGER,INTENT(IN)::N
REAL::FUNC(N)
REAL,INTENT(IN)::X,Y,(:)
! local variables and constants
REAL,PARAMETER::G=9.81
REAL::BIG_D,SMALL_D,F,L,BIG_A,SMALL_A,PI
BIG_D=2.5;SMALL_D=0.8;L=1200.0;F=0.005;PI=4.*ATAN(1.)
BIG_A=PI*BIG_D**2/4.0;SMALL_A=PI*SMALL_D**2/4.0
FUNC(1)=-Y(2)*SMALL_A/BIG_A
FUNC(2)=(Y(1)-4*F*L*Y(2)**2/2./SMALL_D)*G/L
RETURN
END FUNCTION FUNC
END PROGRAM SURGE_TANK
```

The initial conditions for case 1 might be $y(0)=150.0$, $v(0)=10.0$. Under these circumstances, the water level in the surge tank reaches 7 m above reservoir level in about 350 seconds after turbine shut-down. In the second case the initial conditions are $y(0)=0$, $v(0)=0$ and the water level in the tank drops to 70 m below reservoir level about 100 seconds after start-up.

10. The Structure Chart is as given in the text

```
PROGRAM DATE_OF_EASTER
! a program to calculate the date of easter
  INTEGER::NUMBER_OF_YEARS,YEAR_WANTED,DAY
  CHARACTER::*8 MONTH,DESIGNATION
  READ*,NUMBER_OF_YEARS
    DO I=1,NUMBER_OF_YEARS
      READ*,YEAR_WANTED
      CALL EASTER(YEAR_WANTED,MONTH,DAY)
        SELECT CASE (DAY)
        CASE (1,21,31)
          DESIGNATION='ST'
        CASE(2,22)
          DESIGNATION='ND'
        CASE(3,23)
          DESIGNATION='RD'
        CASE DEFAULT
          DESIGNATION='TH'
        END SELECT
      PRINT'(A,I5,A,I3,A,A)','EASTER DAY IN THE YEAR',YEAR_WANTED&
      ,'  FALLS ON',DAY,DESIGNATION,MONTH
    END DO
  CONTAINS
```