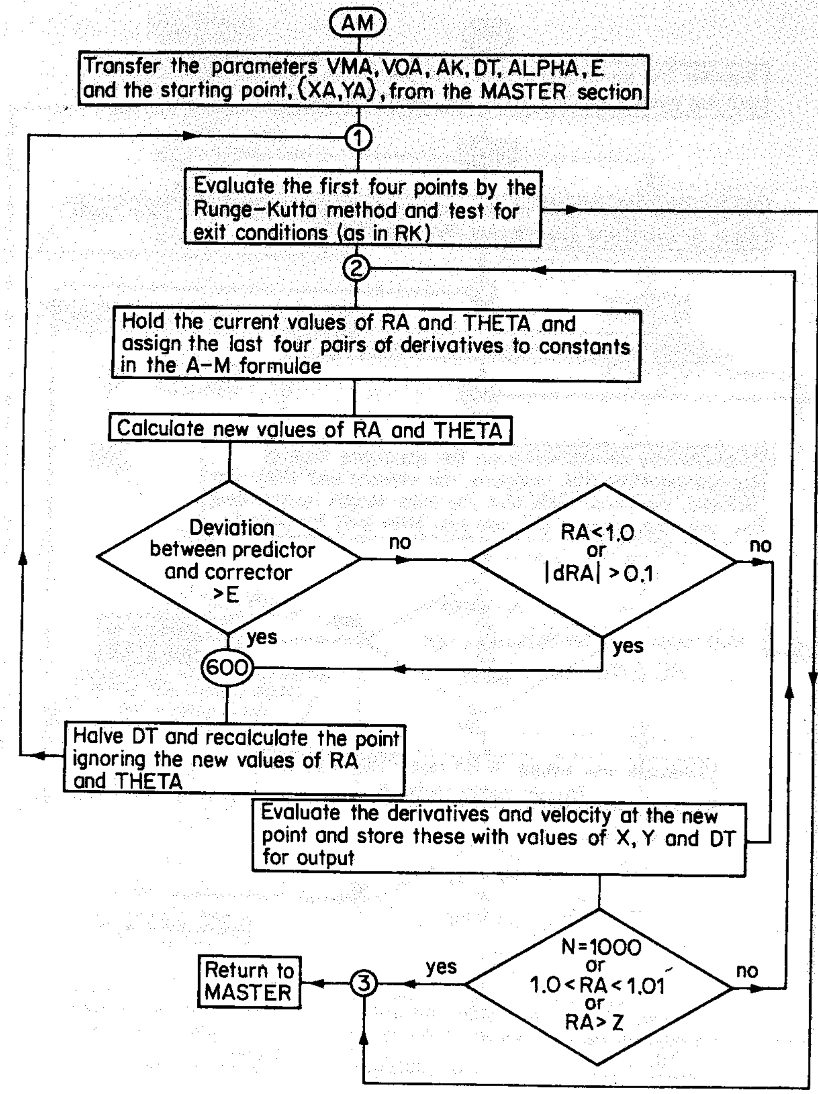


Figure 6. Flow chart of the AM subroutine

the secondary x-axis). The trajectories can be divided into two kinds, differing whether the particle is captured or not. The borderline case of a trajectory for which the particle is just captured is called the critical trajectory. The distance between the parallel part of the critical trajectory and the x-axis defines the capture radius (see Figure 7).

There are two critical trajectories, one above and one below the axis of flow. Thus we have two capture radii, R_{ca1} and R_{ca2}. The average capture radius $\frac{1}{2}(R_{ca1}+R_{ca2})$ is exactly the quantity R_{ca} which appears in equation (1). The proof that this single-wire analysis applies to the multiwire regular matrix can be found in Birss, Gerber, and Parker.[2]

The XSECT subroutine always computes in principle the capture radius, R_{ca1}, above the axis of flow. To obtain R_{ca2}, the computation has to be

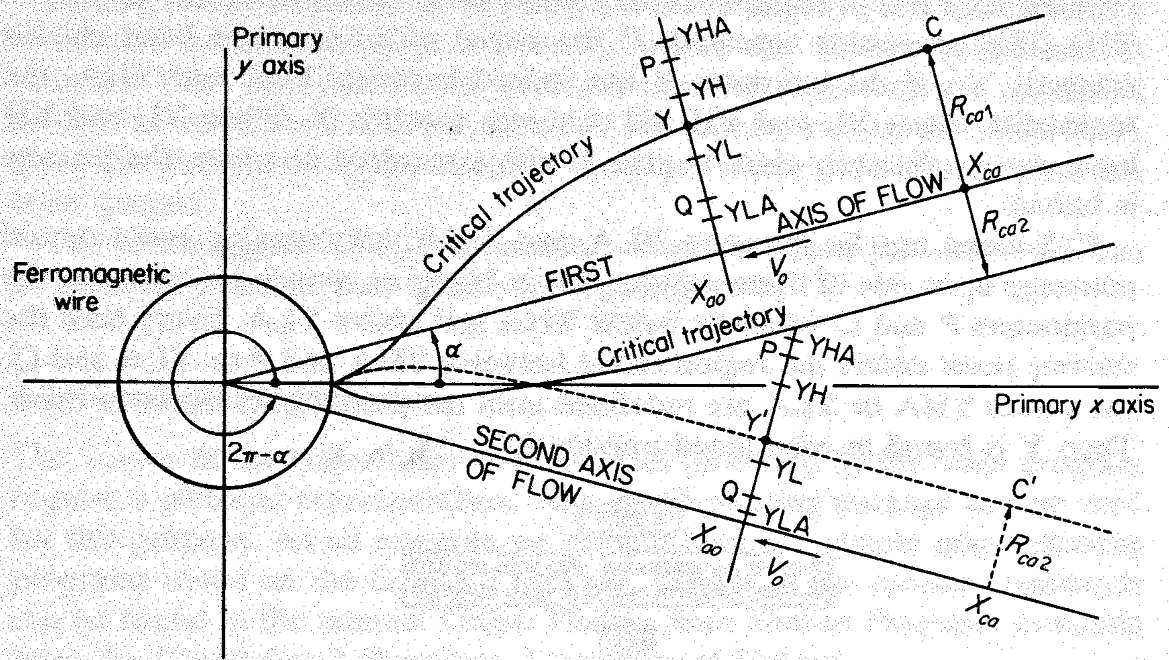


Figure 7. Arrangement of the axes of flow and various parameters used in the XSECT subroutine

performed in the mirror image of the original flow with respect to the primary x-axis. This is achieved by the same input data as for R_{cal}, only instead of α an angle $\beta = 2\pi - \alpha$ is used.

The XSECT subroutine uses the same limiting conditions (i), (iii), and (iv) as the previous subroutines. However, instead of (ii) a more rigorous condition, which would be linked to the ratio $|V_{ma}/V_{0a}|$ and would define the escape of the particle, is required. Bearing in mind that the escape occurs when the fluid drag ultimately exceeds the magnetic traction force, we can find, by analysing (19a), the relation

(ii′)
$$r_a \cos(\theta - \alpha) < -\sqrt[3]{\left|\frac{V_{ma}}{V_{0a}}\right|}$$

which is the condition used in XSECT in place of (ii).

The action of the XSECT subroutine can be described in terms of values YLA, YL, YHA, YH, which are respectively the initial and current estimates of the lower and upper value of Y, the ordinate of a point on the critical trajectory, and in terms of auxiliary parameters P and Q (see Figure 7).

At the beginning YL = YLA and YH = YHA. The particle is started at the point {XAO, YAO = $\frac{1}{2}$(YLA + YHA)} and the trajectory is produced. If the particle is captured, YL is raised to YAO; if it is not captured, YH is lowered to YAO. The new starting point is set, in either case, as $\frac{1}{2}$(YL+YH). This process is repeated. If the initial estimates have been chosen correctly, i.e. if the unknown Y lies indeed between YLA and YHA, the sequential values YL and YH will converge towards Y. When YL and YH have come sufficiently close to give Y with a required accuracy, the process is halted.

If Y does not lie between YLA and YHA, the starting point would converge upon one of these values. This is, however, avoided by defining the parameters *P* and *Q* which lie below YHA and above YLA. Every time the starting point enters the region either between YHA and P or YLA and Q, the values YHA or YLA are redefined until the point Y lies between them. Then Y is found as mentioned previously.

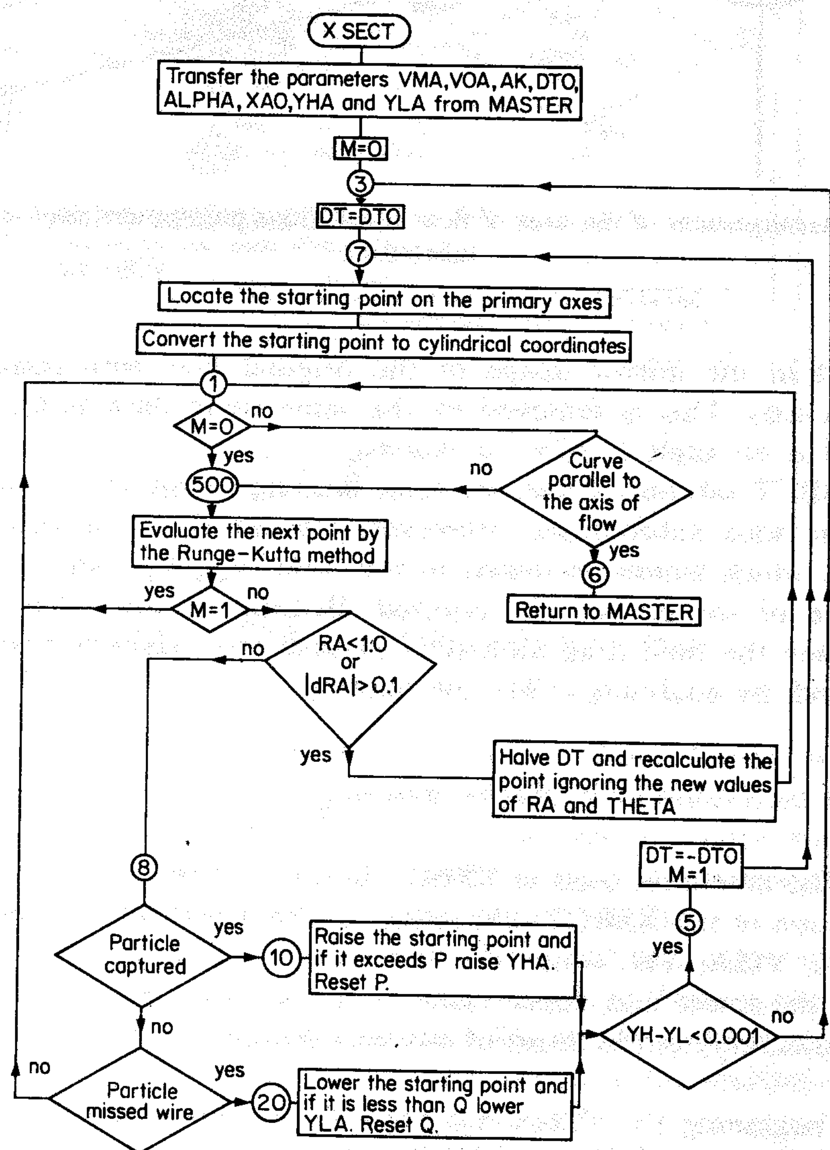


Figure 8. Flow chart of the XSECT subroutine

Having found Y, the time step length is made negative and the trajectory is iterated backwards up to a point C, where the difference between its ordinate YCA and that of the previous iteration is less than a stipulated error. The coordinate YCA is taken as the capture radius, the value XCA gives the distance from the origin to the place where the capture radius has been gauged.

The action of the XSECT subroutine, which has been just described, can be followed from its flow chart shown in Figure 8.

4.2 Graph-plotting programs

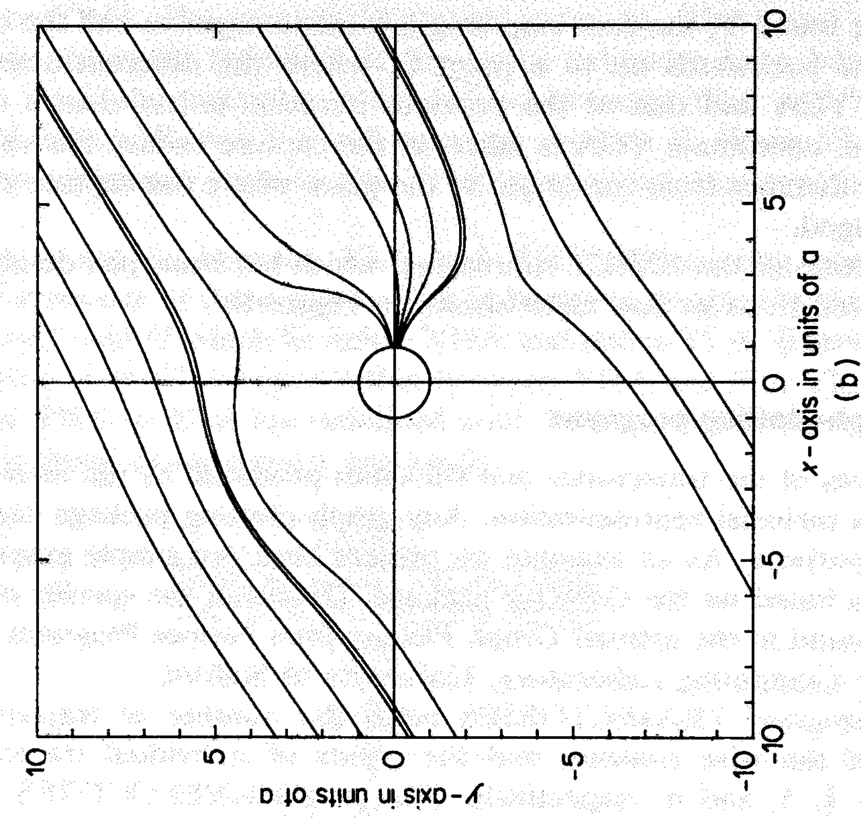
The curves of the trajectories and velocities produced by the main program require a pictorial representation. Any graph-plotting package can be used for this purpose. As an example we present here two simple graph-plotting programs based on the GINO-F package. Details of the specific commands can be found in the manual *Graph Plotting from Fortran Programs* available from the Computing Laboratory, University of Salford.

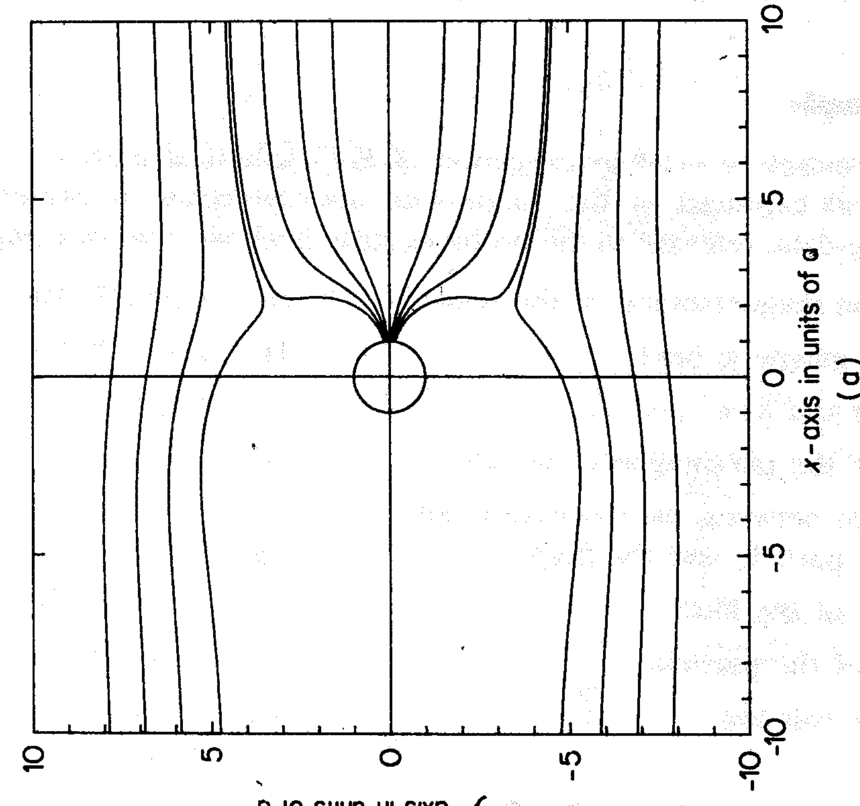
The program TRAJECTORIES reads the number of trajectories, the points of the wire contour, and the points of individual trajectories via channels 1, 5, and 6, respectively. The program VELOCITIES reads the number of velocity curves, the velocity range, the angle α via channel 1, and the points of the individual velocity curves via channel 6.

4.3 Example

As an example a brief investigation of FeO spherical particles carried by water and captured in the single-wire approximation is presented. The following data, relevant to the problem, have been used in the computation:

saturation magnetization of the wire	$M_s = 1.6 \times 10^6\ \mathrm{Am^{-1}}$
external magnetic field	$H_0 = 1.0 \times 10^6\ \mathrm{Am^{-1}}$
radius of the wire	$a = 10\ \mu\mathrm{m}$
radius of the paramagnetic particle	$R = 1\ \mu\mathrm{m}$
difference between the susceptibilities of the particle and the fluid	$\chi = 7.178 \times 10^{-3}$
viscosity of the fluid	$\eta = 1.0 \times 10^{-3}\ \mathrm{Nm^{-2}\,s}$
density of the particle	$\rho_p = 5.7 \times 10^3\ \mathrm{kgm^{-3}}$
magnetic velocity	$V_{ma} = V_m/a = \frac{2}{9}(\chi\mu_0 M_s H_0 R_a^2/\eta) = 3.207 \times 10^4\ \mathrm{s^{-1}}$
fluid velocity	$V_{0a} = -6.288 \times 10^2\ \mathrm{s^{-1}}$


y-axis in units of a
x-axis in units of a
(b)


y-axis in units of a
x-axis in units of a
(a)

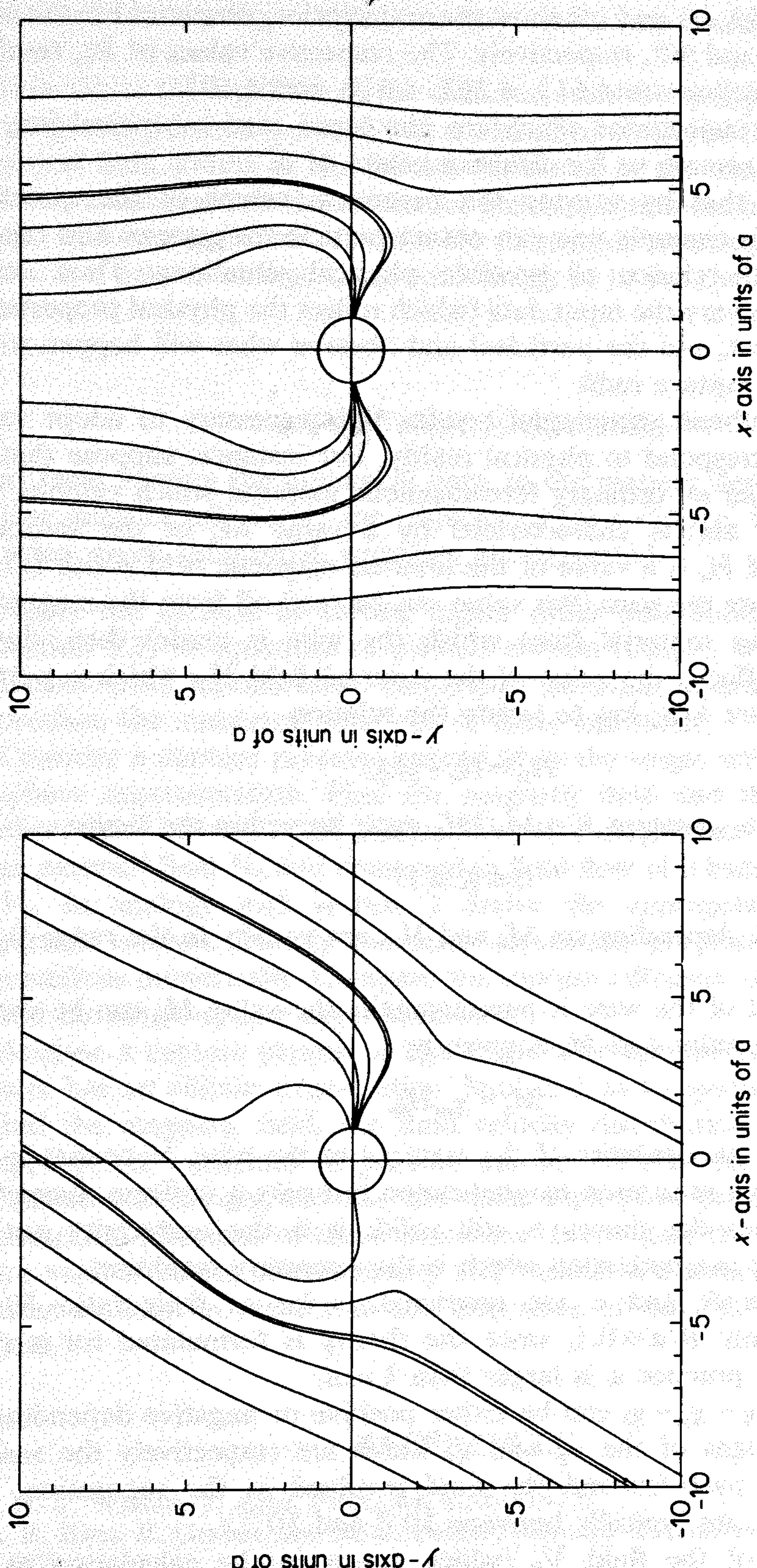


Figure 9. Trajectories of FeO spherical particles for the angles of fluid flow: (a) $\alpha = 0°$, (b) $\alpha = 30°$, (c) $\alpha = 60°$, (d) $\alpha = 90°$

Figures 9(a), (b), (c) and (d) show the trajectories computed for the angle $\alpha=0°$, $30°$, $60°$, and $90°$, respectively. The respective values of R_{ca} resulting from the computation are 4.613, 4.367, 4.410, and 4.697.

Note that the trajectories which are just super- and subcritical for, say, $\alpha=0°$ need not remain so for different values of α.

It is obvious that by varying the numerical values of the quantities mentioned in this example one can obtain particle trajectories and capture radii for a wide choice of possible physical situations. Thus, as an exercise, one can vary the input data (which reflect the physical properties of the wire, the fluid, and the particles) and observe what will happen to the trajectories and capture radii.

However, to obtain meaningful results, it is necessary to adopt values which would correspond to physical reality. For instance, suppose that the wire is made from an ordinary ferromagnetic material which exhibits only small hysteresis and is characterized by a value M_s of the saturation magnetization. If H_s is a value of the internal magnetic field which is large enough to saturate the wire (this value can be read off from the magnetization curve of the material from which the wire is made) then, due to demagnetizing effects, the value of the external field H_0, which is perpendicular to the wire axis, has to satisfy the relation

$$H_0 \geqslant H_s + \tfrac{1}{2}M_s. \tag{36}$$

Consequently, the constant $K=M_s/2H_0$ must lie within the limits

$$0 \leqslant K \leqslant 1. \tag{37}$$

The values of K, depending on M_s and H_0, are usually in the range 0.4 to 0.9.

If the material of the wire is paramagnetic, the value H_0 can be chosen without any constraint and M_s is given by

$$M_s = \chi_m H_0, \tag{38}$$

where χ_m is the susceptibility of the material of the wire. Note that in this case M_s is not the saturation magnetization but only a uniform magnetization. Nevertheless the theory is still valid; it is the uniformity not the saturation of the magnetization which is the necessary condition.

The quantities R and a are restricted as far as their ratio R/a is concerned, namely $R/a \leqslant 0.1$, since the theory is formulated for a small particle limit. In practice a is larger than 1 μm.

The quantity $\chi=\chi_p-\chi_f$ can be either positive or negative depending on the value and signs of the χ_p and χ_f which are respectively the susceptibilities of the particles and the fluid involved in the separation. The magnitudes of χ are typically between 10^{-5} and 10^{-2}.

The velocity of the fluid V_0 (which enters in the calculation as the normalized velocity $V_{0a} = V_0/\acute{a}$) can vary over a very broad range, say, from 1 mm s^{-1} to 50 mm s^{-1}.

To facilitate the legibility of the data and results, the input and output in our programs (sections 4.1 and 4.2) are handled simply in the F-format. The numerical field widths of the format stipulated in the programs should be sufficient to accomodate the results for the data within the indicated range of values. If, however, some results were to require an extension of the field widths, it is easy to change them accordingly or to use the E- instead of the F- format.

5. FURTHER EXERCISES

Apart from varying the numerical data, in the manner described in section 4.3 (for instance calculating the trajectories for $\chi < 0$, i.e. for $V_{ma} < 0$), the following, more substantial, exercises are suggested.

(1) Apply the method of solving higher order differential equations (see section 3.3) to equations of particle motion (18a, b), which contain the second-order inertial term. Generalize the main separation program to produce the numerical solutions of these equations.

(2) Consider a capture process, instead of in the single wire, in the single-sphere approximation. Find the magnetic field and the fluid velocity distribution around a ferromagnetic sphere magnetized to saturation by an external field H_0 and immersed in fluid flow of a background velocity V_0. In analogy with section 2 derive the appropriate equations of particle motion. Modify the main separation program to solve these equations numerically. Compare the capture efficiency of the single wire and the single-sphere approximations.

(3) Analyse a capture process in the single-wire approximation, where the wire has an elliptic cross-section. Similarly, as suggested in section 5.2, find the magnetic field, the fluid velocity distribution, and derive the appropriate equations of particle motion. Modify the FUNCTN subroutine accordingly and use the main separation program for finding numerical solutions of these equations. Investigate the capture process for various values of eccentricity and various orientations of the elliptic cross-sections in respect to the directions of fluid flow and magnetic field.

REFERENCES

1. J. H. P. Watson, *J. Appl. Phys.*, **44,** 4209 (1973).
2. R. R. Birss, R. Gerber, and M. R. Parker, *Filtration & Separation*, **July/August,** 1 (1977).

3. R. R. Birss, R. Gerber, and M. R. Parker, *I.E.E.E. Trans. Magn.*, **MAG-12,** 892 (1976).
4. B. I. Bleaney and B. Bleaney, *Electricity and Magnetism*, 2nd edn. (Clarendon Press, Oxford) 47 (1965).
5. W. F. Lawson Jr., W. H. Simons, and R. P. Treat, *J. Appl. Phys.*, **48,** 3213 (1977).
6. D. M. Young and R. T. Gregory, *A Survey of Numerical Mathematics* (Addison-Wesley, Reading, Mass.) 422 (1972).

TYPICAL INPUT DATA FOR THE MAIN PROGRAM (FOR ALPHA = 30 DEGR.)

```
  2    1    16    0.5236
 15.0                   1.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   2.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   3.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   4.48
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   4.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   5.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   6.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                   7.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -1.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -2.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -3.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -4.48
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -4.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -5.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -6.6
 32071.0     -628.8     0.8      0.0002     0.01
 15.0                  -7.6
 32071.0     -628.8     0.8      0.0002     0.01
```

TYPICAL OUTPUT FROM THE MAIN PROGRAM (FOR ALPHA = 30 DEGR.)

K=.80 VMA= 32071.00 VOA= -628.80
VMA/VOA= -51.00 ALPHA=0.523600
XAO= 15.0000
YAO= 1.6000
NUMBER OF COMPUTED POINTS Q= 219

X	Y	V	DRA/DT	DTH/DT
12.19037	8.88566	628.09008	-625.39780	3.85099
12.13641	8.85351	628.08329	-625.37877	3.87576
12.08246	8.82135	628.07635	-625.35959	3.90071
12.02852	8.78918	628.06925	-625.34026	3.92585
11.97458	8.75701	628.06199	-625.32078	3.95118
11.92065	8.72482	628.05457	-625.30115	3.97669
11.86672	8.69263	628.04698	-625.28136	4.00238
11.81280	8.66042	628.03921	-625.26142	4.02827
11.75889	8.62820	628.03127	-625.24132	4.05433
11.70499	8.59598	628.02315	-625.22107	4.08058
11.65109	8.56374	628.01484	-625.20066	4.10702
11.59720	8.53149	628.00634	-625.18009	4.13364
11.54332	8.49923	627.99764	-625.15935	4.16045
11.48944	8.46697	627.98873	-625.13846	4.18744
11.43558	8.43468	627.97962	-625.11740	4.21462
11.38172	8.40239	627.97030	-625.09618	4.24198
11.32787	8.37009	627.96075	-625.07479	4.26952
11.27403	8.33777	627.95098	-625.05324	4.29724
11.22019	8.30545	627.94097	-625.03151	4.32515
11.16637	8.27311	627.93073	-625.00962	4.35323
11.11255	8.24075	627.92024	-624.98757	4.38149
11.05874	8.20839	627.90950	-624.96533	4.40993
11.00495	8.17601	627.89850	-624.94293	4.43854
10.95116	8.14362	627.88723	-624.92036	4.46733
10.89738	8.11121	627.87569	-624.89761	4.49628
10.84361	8.07879	627.86387	-624.87469	4.52541
10.78985	8.04636	627.85176	-624.85159	4.55470
10.73610	8.01391	627.83935	-624.82832	4.58416
10.68236	7.98145	627.82664	-624.80487	4.61378
10.62863	7.94898	627.81362	-624.78124	4.64356
10.57491	7.91649	627.80028	-624.75744	4.67349
10.52120	7.88398	627.78660	-624.73345	4.70358
10.46750	7.85146	627.77259	-624.70929	4.73381
10.41382	7.81892	627.75822	-624.68495	4.76418
10.36014	7.78636	627.74350	-624.66043	4.79470
10.30648	7.75379	627.72842	-624.63573	4.82535
10.25283	7.72121	627.71295	-624.61085	4.85613
10.19919	7.68860	627.69710	-624.58579	4.88703
10.14556	7.65598	627.68085	-624.56055	4.91806
10.09195	7.62334	627.66418	-624.53513	4.94919
10.03835	7.59068	627.64710	-624.50953	4.98043
9.98476	7.55800	627.62959	-624.48375	5.01177
9.93119	7.52531	627.61164	-624.45779	5.04320
9.87763	7.49259	627.59323	-624.43166	5.07472

TYPICAL OUTPUT FROM THE MAIN PROGRAM (FOR ALPHA = 30 DEGR.)

9.82409	7.45986	627.57436	-624.40535	5.10631
9.77056	7.42710	627.55500	-624.37886	5.13796
9.71704	7.39433	627.53516	-624.35220	5.16968
9.66354	7.36153	627.51481	-624.32537	5.20144
9.61006	7.32872	627.49394	-624.29837	5.23323
9.55659	7.29588	627.47255	-624.27120	5.26506
9.50313	7.26302	627.45061	-624.24386	5.29689
9.44970	7.23013	627.42811	-624.21636	5.32872
9.39628	7.19722	627.40504	-624.18870	5.36055
9.34288	7.16429	627.38138	-624.16088	5.39234
9.28949	7.13134	627.35713	-624.13291	5.42409
9.23612	7.09836	627.33225	-624.10479	5.45578
9.18278	7.06536	627.30675	-624.07652	5.48740
9.12945	7.03233	627.28060	-624.04811	5.51893
9.07614	6.99927	627.25379	-624.01956	5.55034

AND SO ON UNTIL THE END OF THE CURVE:

1.99124	0.41735	5001.67286	-4677.03717	-871.26219
1.93562	0.38296	5480.27835	-5160.53217	-934.83109
1.87304	0.34697	6097.52330	-5784.10413	-1013.04001
1.80132	0.30904	6929.73002	-6624.48599	-1112.93695
1.71689	0.26863	8125.14141	-7830.53890	-1247.62033
1.66812	0.24723	8948.72743	-8660.67437	-1335.55755
1.61342	0.22484	10016.44810	-9736.00489	-1444.80580
1.55086	0.20120	11465.39899	-11193.90579	-1586.01225
1.47731	0.17593	13564.56137	-13303.78436	-1779.20508
1.38701	0.14839	16933.11739	-16685.52037	-2068.28761
1.33225	0.13340	19551.81429	-19312.12550	-2279.52085
1.26754	0.11726	23409.29863	-23178.74971	-2574.57313
1.18736	0.09943	29766.09748	-29546.41012	-3029.53802
1.13808	0.08959	34867.61021	-34654.26075	-3373.57563
1.07897	0.07882	42661.96776	-42455.80182	-3872.16617
1.00376	0.06665	56335.52544	-56137.70355	-4688.95543

THEN THE SAME OUTPUT FORMAT FOR CURVES NO.:2,3,4,.....,16.

MAIN MAGNETIC SEPARATION PROGRAM

```
C     THE MAIN PROGRAM SERVES AS A CARRIAGE TO INPUT DATA AND OUTPUT RESULTS
C  AFTER THE EQUATIONS HAVE BEEN SOLVED.IT ALSO DIRECTS THE DATA TO THE
C  APPROPRIATE SUBROUTINE THAT IS TO PERFORM THE ITERATIONS ON THE EQUATIONS
C  WHICH ARE HELD IN THE SUBROUTINE 'FUNCTN'.THERE ARE TWO METHODS OF
C  ITERATION,THE RUNGE-KUTTA AND THE ADAMS-MOULTON METHOD.
C  IN ADDITION TO THIS THE SUBROUTINE 'XSECT' WILL FIND THE CAPTURE CROSS-
C  SECTION OF THE WIRE FOR DIFFERENT FLUID VELOCITIES.
C
C     VMA,VOA,AK AND DTO ARE THE MAGNETIC AND FLUID VELOCITIES,
C     THE SHORT RANGE CONSTANT AND THE TIME STEP,RESPECTIVELY.
C     E IS THE MAXIMUM DEVIATION BETWEEN PREDICTOR AND CORRECTOR IN THE
C     PREDICTOR-CORRECTOR METHODS.
C     YLA AND YHA ARE PARAMETERS IN 'XSECT' SUBROUTINE.
C     J IS THE CONTROL INTEGER WHICH SELECTS THE METHOD OF ITERATION.
C     J=1,2,3.
C     K IS THE CONTROL INTEGER WHICH SELECTS THE MODE OF OUTPUT.
C     K=1,2,3.
C     IO IS THE NUMBER OF INITIAL VALUES OF X AND Y,READ IN AS XA AND YA,
C     FOR WHICH CURVES WILL BE PRODUCED.
C     Z IS THE RATIO OF VMA TO VOA.
C     YC IS THE CAPTURE CROSS-SECTION.
C     V IS THE VELOCITY OF THE PARTICLE AT A GIVEN POINT.
C     THE SUBROUTINES WILL PRODUCE Q POINTS(I.E. Q X-Y PAIRS)FOR
C     EACH CURVE.THIS PARAMETER IS IMPORTANT IN THE PLOTTING OF THE
C     CURVES.
C     DAR,DIT AND DEET ARE ARRAYS OF THE GRADIENTS, I.E. THE RADIAL
C     AND ANGULAR VELOCITIES, AND THE TIME STEP,RESPECTIVELY.
C     ALPHA IS THE INCLINATION OF THE FLUID VELOCITY TO THE X-AXIS.
C
      INTEGER Q
      DIMENSION X(1000),Y(1000),G(1000),F(1000),V(1000)
      DIMENSION DAR(1000),DIT(1000),DEET(1000)
      READ(1,1001)J,K,IO,ALPHA
      I=0
  100 I=I+1
      IF(J.EQ.3)GOTO 3
      READ(1,1002)XAO,YAO
      READ(1,1000)VMA,VOA,AK,DTO,E
      XA=XAO*COS(ALPHA)-YAO*SIN(ALPHA)
      YA=YAO*COS(ALPHA)+XAO*SIN(ALPHA)
      ZZ=VMA/VOA
      DT=DTO
      GOTO(1,2),J
    1 CALL RK(XA,YA,VMA,VOA,AK,DT,NN,X,Y,V,ALPHA,DAR,DIT,DEET)
      GOTO 10
    2 CALL AM(XA,YA,VMA,VOA,AK,DT,NN,X,Y,V,E,ALPHA,DAR,DIT,DEET)
      GOTO 10
    3 READ(1,1004)VMA,VOA,AK,DT,YLA,YHA,XAO,ALPHA
      CALL XSECT(YLA,YHA,VMA,VOA,AK,DT,Z,YC,XC,XAO,ALPHA)
      IF(K.EQ.3)GOTO4
      WRITE(2,1006)AK,VMA,VOA,Z,ALPHA,XC,YC
      IF(K.EQ.2)GOTO5
    4 WRITE(3,1005)Z,YC,XC,ALPHA
    5 IF(I.EQ.IO)GOTO 20
      I=I+1
      GOTO 3
```

MAIN MAGNETIC SEPARATION PROGRAM

```
   10 Q=NN
      IF(K.EQ.3)GOTO6
      WRITE(2,1009) AK,UMA,VOA,ZZ,ALPHA,XAO,YAO
      WRITE(2,1007)Q
      WRITE(2,1008)
      WRITE(2,1002)(X(N),Y(N),V(N),DAR(N),DIT(N),DEET(N),N=1,Q)
      WRITE(2,1010)
      IF(K.EQ.2)GOTO7
    6 WRITE(3,1003)Q
      WRITE(3,1002)(X(N),Y(N),V(N),DAR(N),DIT(N),DEET(N),N=1,Q)
    7 IF(I.EQ.IO)GOTO 20
      GOTO 100
   20 STOP
 1000 FORMAT(2F10.2,F10.5,2F10.8,2F10.5)
 1001 FORMAT(3I5,F10.5)
 1002 FORMAT(1X,F10.5,2F20.5,2F15.5,F15.10)
 1003 FORMAT(1X,I10)
 1004 FORMAT(2F10.2,F10.5,F10.8,5F10.5)
 1005 FORMAT(1X,F10.5,3F20.5)
 1006 FORMAT(5H0   K=,F3.2,7H    VMA=,F10.2,
     C7H    VOA=,F10.2/10H  VMA/VOA=,
     CF7.2,9H    ALPHA=,F8.6/7H    XCA=,
     CF9.4/7H    RCA=,F9.4)
 1007 FORMAT(1X,31H  NUMBER OF COMPUTED POINTS   Q=,I4///)
 1008 FORMAT(1X,88H      X                     Y                    V
     C    DRA/DT             DTH/DT         DT///)
 1009 FORMAT(5H0   K=,F3.2,7H    VMA=,F10.2,
     C7H    VOA=,F10.2/10H  VMA/VOA=,
     CF7.2,9H    ALPHA=,F8.6/7H    XAO=,
     CF9.4/7H    YAO=,F9.4)
 1010 FORMAT(////)
      END
C
C
C  THE FUNCTN SUBROUTINE.
C
C  THIS SUBROUTINE CONTAINS THE EQUATIONS TO BE ITERATED.
C
      SUBROUTINE FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      DRADT=VOA*(1.0-(RA**(-2)))*COS(THETA-ALPHA)-VMA*AK/(RA**5)-VMA*
     1COS(2.0*THETA)/(RA**3)
      DTHDT=-VOA*(1.0+(RA**(-2)))*SIN(THETA-ALPHA)/RA-VMA*SIN(2.0*THETA
     1)/(RA**4)
      RETURN
      END
C
C
C  THE RUNGE-KUTTA METHOD.
C
C  THIS IS A SELF-STARTING ,FOURTH-ORDER METHOD OF HIGH ACCURACY.
C  IT INCORPORATES THE STEP LENGTH ADJUSTMENT.
C
      SUBROUTINE RK(XA,YA,VMA,VOA,AK,DT,N,X,Y,V,ALPHA,DAR,DIT,DEET)
      REAL L,KR1,KT1,KR2,KT2,KR3,KT3,KR4,KT4
      DIMENSION X(1000),Y(1000),V(1000)
      DIMENSION DAR(1000),DIT(1000),DEET(1000)
```

```
      N=1
      X(N)=XA
      Y(N)=YA
      RA=SQRT(XA*XA+YA*YA)
      THETA=ATAN2(YA,XA)
      Z=RA+0.2
      IF(Z.LE.14.2)Z=14.2
    1 R=RA
      TH=THETA
   40 CALL FUNCTN(RA,THETA,UMA,VOA,AK,DRADT,DTHDT,ALPHA)
      V(N)=SQRT(DRADT*DRADT+RA*RA*DTHDT*DTHDT)
      DAR(N)=DRADT
      DIT(N)=DTHDT
      DEET(N)=DT
      IF(N.EQ.1000)GOTO 2
      IF(RA.LT.1.01.AND.RA.GT.1.00)GOTO 2
      IF(RA.GT.Z)GOTO 2
      KR1=DRADT*DT
      KT1=DTHDT*DT
      RA=R+0.5*KR1
      THETA=TH+0.5*KT1
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR2=DRADT*DT
      KT2=DTHDT*DT
      RA=R+0.5*KR2
      THETA=TH+0.5*KT2
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR3=DRADT*DT
      KT3=DTHDT*DT
      RA=R+KR3
      THETA=TH+KT3
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR4=DRADT*DT
      KT4=DTHDT*DT
      DRA=(KR1+2.0*(KR2+KR3)+KR4)/6.0
      DTH=(KT1+2.0*(KT2+KT3)+KT4)/6.0
      RA=R+DRA
      THETA=TH+DTH
      IF(RA.LT.1.00.OR.ABS(DRA).GT.0.1)GOTO 4
      N=N+1
      X(N)=RA*COS(THETA)
      Y(N)=RA*SIN(THETA)
      GOTO 1
    4 DT=DT/2.0
      RA=R
      THETA=TH
      GOTO 40
    2 RETURN
      END
C
C
C  THE ADAMS-MOULTON METHOD.
C
C  THIS METHOD USES THE FIFTH-ORDER ADAMS-MOULTON PREDICTOR-
C  CORRECTOR METHOD UTILISING THE RUNGE-KUTTA METHOD AS ITS STARTER
C  TO FIND THE FIRST FOUR POINTS.
```

MAIN MAGNETIC SEPARATION PROGRAM

```
C   THE ERROR IN THIS TECHNIQUE IS GOVERNED BY E.
C   IT ALSO INCORPORATES THE STEP LENGTH ADJUSTMENT
C
      SUBROUTINE AM(XA,YA,VMA,VOA,AK,DT,N,X,Y,V,E,ALPHA,F,G,DEET)
      REAL L,KR1,KT1,KR2,KT2,KR3,KT3,KR4,KT4
      DIMENSION X(1000),Y(1000),F(1000),G(1000),V(1000)
      DIMENSION DEET(1000)
      NO=4
      N=1
      X(N)=XA
      Y(N)=YA
      RA=SQRT(XA*XA+YA*YA)
      THETA=ATAN2(YA,XA)
      Z=RA+0.2
      IF(Z.LE.14.2)Z=14.2
C   START OF RUNGE-KUTTA
    1 R=RA
      TH=THETA
   40 CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      V(N)=SQRT(DRADT*DRADT+RA*RA*DTHDT*DTHDT)
      DEET(N)=DT
      F(N)=DRADT
      G(N)=DTHDT
      IF(N.EQ.1000)GOTO 3
      IF(RA.LE.1.01.AND.RA.GE.1.00)GOTO 3
      IF(RA.GE.Z)GOTO 3
      IF(N.GE.NO)GOTO 2
      KR1=DRADT*DT
      KT1=DTHDT*DT
      RA=R+0.5*KR1
      THETA=TH+0.5*KT1
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR2=DRADT*DT
      KT2=DTHDT*DT
      RA=R+0.5*KR2
      THETA=TH+0.5*KT2
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR3=DRADT*DT
      KT3=DTHDT*DT
      RA=R+KR3
      THETA=TH+KT3
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR4=DRADT*DT
      KT4=DTHDT*DT
      DRA=(KR1+2.0*(KR2+KR3)+KR4)/6.0
      DTH=(KT1+2.0*(KT2+KT3)+KT4)/6.0
      RA=R+DRA
      THETA=TH+DTH
      IF(RA.LT.1.00.OR.ABS(DRA).GT.0.1)GOTO 4
      N=N+1
      X(N)=RA*COS(THETA)
      Y(N)=RA*SIN(THETA)
      GOTO 1
    4 DT=DT/2.0
      NO=N+4
      RA=R
```

MAIN MAGNETIC SEPARATION PROGRAM

```
      THETA=TH
      GOTO 40
C  START OF ADAMS-MOULTON
    2 R=RA
      TH=THETA
      F1=F(N)
      G1=G(N)
      N=N-1
      F2=F(N)
      G2=G(N)
      N=N-1
      F3=F(N)
      G3=G(N)
      N=N-1
      F4=F(N)
      G4=G(N)
      N=N+3
      AMR1=R+DT*(55.0*F1-59.0*F2+37.0*F3-9.0*F4)/24.0
      AMT1=TH+DT*(55.0*G1-59.0*G2+37.0*G3-9.0*G4)/24.0
      RA=AMR1
      THETA=AMT1
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      AMR2=R+DT*(9.0*DRADT+19.0*F1-5.0*F2+F3)/24.0
      AMT2=TH+DT*(9.0*DTHDT+19.0*G1-5.0*G2+G3)/24.0
      ETA=ABS(AMR1-AMR2)/AMR2
      IF(ETA.GE.E)GOTO 600
      IF(AMT2.LT.1.0E-10)GOTO 700
      ZETA=ABS(AMT1-AMT2)/AMT2
      IF(ZETA.GE.E)GOTO 600
      GOTO 700
  600 DT=DT/2.0
      NO=N+4
      RA=R
      THETA=TH
      GOTO 1
  700 RA=AMR2
      THETA=AMT2
      IF(RA.LT.1.00.OR.ABS(RA-R).GT.0.1)GOTO 600
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      N=N+1
      F(N)=DRADT
      G(N)=DTHDT
      V(N)=SQRT(DRADT*DRADT+RA*RA*DTHDT*DTHDT)
      DEET(N)=DT
      X(N)=RA*COS(THETA)
      Y(N)=RA*SIN(THETA)
      IF(N.EQ.1000)GOTO 3
      IF(RA.LE.1.01.AND.RA.GE.1.00)GOTO 3
      IF(RA.GT.2)GOTO 3
      GOTO 2
    3 RETURN
      END
C
C
C  THE XSECT SUBROUTINE.
C
```

MAIN MAGNETIC SEPARATION PROGRAM

```
C  THIS SUBROUTINE CALCULATES THE CAPTURE CROSS SECTION OF THE WIRE.
C  IT DOES SO BY ITERATING A CURVE AND ASSESING WHETHER
C  CAPTURE HAS TAKEN PLACE OR NOT.IF IT HAS, THE STARTING POINT WAS
C  TOO LOW AND IT IS THEN RAISED TO HALF-WAY BETWEEN ITS PRESENT
C  POSITION AND THE STARTING VALUE OF THE CURVE WHEN THE PARTICLE LAST
C  MISSED THE WIRE WHICH IS A LITTLE TOO HIGH.SIMILARLY THE HIGH VALUE
C  IS LOWERED IF THE PARTICLE MISSES THE WIRE.BY REPEATING THIS
C  PROCESS UNTIL THE HIGH AND LOW STARTING POINTS CONVERGE THE CAPTURE
C  CROSS-SECTION IS FOUND.THIS IS REPEATED AS  A FUNCTION OF THE
C  RATIO VMA:VOA.
C
      SUBROUTINE XSECT(YLA,YHA,VMA,VOA,AK,DTO,Z,YC,XC,XAO,ALPHA)
      REAL L,KR1,KT1,KR2,KT2,KR3,KT3,KR4,KT4
      WRITE(2,1040)
 1040 FORMAT(13H0  ITERATIONS//)
      M=0
      D=0.0
      Z=VMA/VOA
      YAO=(YLA+YHA)/2.0
      YH=YHA
      YL=YLA
    3 DT=DTO
    7 YA=XAO*SIN(ALPHA)+YAO*COS(ALPHA)
      XA=XAO*COS(ALPHA)-YAO*SIN(ALPHA)
      RA=SQRT(XA*XA+YA*YA)
      THETA=ATAN2(YA,XA)
    1 R=RA
      TH=THETA
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      IF(M.EQ.0)GOTO 500
      YC=RA*SIN(THETA-ALPHA)
      XC=RA*COS(THETA-ALPHA)
      DRC=ABS(YC-D)
      IF(DRC.LT.0.0005)GOTO 6
      D=YC
  500 KR1=DRADT*DT
      KT1=DTHDT*DT
      RA=R+0.5*KR1
      THETA=TH+0.5*KT1
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR2=DRADT*DT
      KT2=DTHDT*DT
      RA=R+0.5*KR2
      THETA=TH+0.5*KT2
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR3=DRADT*DT
      KT3=DTHDT*DT
      RA=R+KR3
      THETA=TH+KT3
      CALL FUNCTN(RA,THETA,VMA,VOA,AK,DRADT,DTHDT,ALPHA)
      KR4=DRADT*DT
      KT4=DTHDT*DT
      DRA=(KR1+2.0*(KR2+KR3)+KR4)/6.0
      DTH=(KT1+2.0*(KT2+KT3)+KT4)/6.0
      RA=R+DRA
      THETA=TH+DTH
```

MAIN MAGNETIC SEPARATION PROGRAM

```
      IF(M.EQ.1)GOTO 1
      IF(RA.GT.1.00.AND.ABS(DRA).LT.0.1)GOTO 8
      RA=R
      THETA=TH
      DT=DT/2.0
      GOTO 1
    8 IF(RA.GT.1.00.AND.RA.LT.1.01)GOTO 10
      RL=-(ABS(Z)**0.3333)
      IF(RA*COS(THETA-ALPHA).LT.RL)GOTO 20
      GOTO 1
   10 YL=YAO
      P=YHA-0.1
      WRITE(2,1020)YAO
 1020 FORMAT(1X,F20.5)
      IF(YL.LT.P)GOTO 100
      YHA=YHA+0.2
      YH=YHA
  100 YAO=(YH+YL)/2.0
      W=YH-YL
      IF(W.LT.0.001)GOTO 5
      GOTO 3
   20 YH=YAO
      Q=YLA+0.1
      WRITE(2,1030)YAO
 1030 FORMAT(1X,F30.5)
      IF(YH.GT.Q)GOTO 200
      YLA=YLA/2.0
      IF(YLA.LT.0.05)YLA=0.0
      YL=YLA
  200 YAO=(YH+YL)/2.0
      W=YH-YL
      IF(W.LT.0.001)GOTO 5
      GOTO 3
    5 DT=-DTO
      M=1
      WRITE(2,1010)YAO
 1010 FORMAT(1X,F10.5//)
      GOTO 7
    6 RETURN
      END
```