DIVISION DE ESTUDIOS DE POSGRADO
FACULTAD DE INGENIERIA U. N. A. M.

COMPUTER SUBPROGRAMS FOR THE COMPUTATION OF THE SCREW PARAMETERS OF A RIGID-BODY MOTION

Jorge Angeles *

* *Profesor de Ingeniería Mecánica de la División de Estudios de Posgrado de la U.N.A.M.*

COMPUTER SUBPROGRAMS FOR THE COMPUTATION OF THE SCREW PARAMETERS OF A RIGID-BODY MOTION

Jorge Angeles[1]

INTRODUCTION

Two computer subprograms, SCREW and INSCRU, are contained here. The first one computes the screw parameters of a rigid-body motion with finite separation; the second one, in turn, computes the corresponding parameters when the motion is specified through two infinitesimally separated positions.

Each subprogram is self-contained, its comments describing in detail each stage of the computational procedure. SCREW is based upon [1][2], whereas INSCRU is based upon [2]. In each case, a main program is included, in order to illustrate how these subprograms can be used. The subprograms are written in the FORTRAN IV version of the Burroughs 6700 computer, of the computing center of the University of Mexico (UNAM). Subsidiary subroutines are also included.

The editing, debugging and testing of the subprograms was carried on by Mr. Angel Rojas, a student of the Graduate Division of the Faculty of Engineering, UNAM.

---

[1]Professor of Mechanical Engineering, National Autonomous University of Mexico, C. Universitaria, Apdo. Postal 70-256, México, 20, D.F. México.

[2]Numbers in brackets designate references at the and of the paper

```
C     THIS PROGRAM COMPUTES THE SCREW PARAMETERS OF GIVEN RIGID BODY
C     MOTIONS. THE MOTIONS ARE SPECIFIED THROUGH THE INITIAL AND FINAL
C     POSITIONS OF THREE NONCOLLINEAR POINTS.
C
      REAL AIN(3),BIN(3),CIN(3),AFIN(3),BFIN(3),CFIN(3),E(3),RHO(3)
      COMMON ZERO
      RAD=180.0/(4.0*ATAN(1.0))
      NUM=0
 1    READ(5,/,END=3) N
      ZERO=0.00001
      NUM=NUM+1
      DO 2 I=1,N
           READ(5,/)AIN(I),BIN(I),CIN(I)
           READ(5,/)AFIN(I),BFIN(I),CFIN(I)
 2    CONTINUE
      WRITE(6,9)NUM
      WRITE(6,10)
      WRITE(6,11)(AIN(I),I=1,N)
      WRITE(6,12)(BIN(I),I=1,N)
      WRITE(6,13)(CIN(I),I=1,N)
      WRITE(6,14)(AFIN(I),I=1,N)
      WRITE(6,15)(BFIN(I),I=1,N)
      WRITE(6,16)(CFIN(I),I=1,N)
      CALL SCREW (AIN,BIN,CIN,AFIN,BFIN,CFIN,E,RHO,THETA,DISPL)
      IF(ZERO.LT.0.0) GO TO 1
      WRITE(6,18)(E(I),I=1,N)
      WRITE(6,19)(RHO(I),I=1,N)
      THETA=THETA*RAD
      WRITE(6,20)THETA
      WRITE(6,21)DISPL
      GO TO 1
 3    CALL EXIT
 9    FORMAT(3(/),9X,'EXAMPLE',I3/)
 10   FORMAT(9X,'POINT',20X,'COORDINATES'/22X,'X',16X,'Y',16X,'Z'/)
 11   FORMAT(9X,'A(1)= ',F12.5,5X,F12.5,5X,F12.5)
 12   FORMAT(9X,'B(1)= ',F12.5,5X,F12.5,5X,F12.5)
 13   FORMAT(9X,'C(1)= ',F12.5,5X,F12.5,5X,F12.5/)
 14   FORMAT(9X,'A(2)= ',F12.5,5X,F12.5,5X,F12.5)
 15   FORMAT(9X,'B(2)= ',F12.5,5X,F12.5,5X,F12.5)
 16   FORMAT(9X,'C(2)= ',F12.5,5X,F12.5,5X,F12.5/)
 18   FORMAT(15X,'THE SCREW AXIS HAS THE FOLLOWING DIRECTION COSINES'/
     -       15X,'(VECTOR E) :',F12.5,5X,F12.5,5X,F12.5)
 19   FORMAT(15X,'THE POINT ON THE SCREW AXIS CLOSEST TO THE ORIGIN ',
     -       'HAS'/15X,'THE FOLLOWING X-,Y-AND Z COORDINATES'/
     -       15X,'(VECTOR RHO) :',F12.5,5X,F12.5,5X,F12.5)
 20   FORMAT(15X,'THE ANGLE OF ROTATION (THETA) IS',F12.5,' DEGREES')
 21   FORMAT(15X,'THE DISPLACEMENT ALONG THE SCREW AXIS (DISPL) IS',
     -       F12.5/)
      END
```

```
#WAITING FOR WORKER
      SUBROUTINE SCREW(AIN,BIN,CIN,AFIN,BFIN,CFIN,E,RHO,THETA,DISPL)
C
C  THIS SUBROUTINE COMPUTES THE SCREW-PARAMETERS OF A RIGID BODY MOTION.
C
C
C  INPUT:
C   THE X, Y AND Z-COORDINATES OF THREE NONCOLLINEAR POINTS OF THE
C   RIGID BODY IN BOTH ITS INITIAL (AIN,BIN,CIN 3-DIMENSIONAL VECTORS)
C   AND  FINAL (AFIN,BFIN,CFIN 3-DIMENSIONAL VECTORS) CONFIGURATIONS.
C
C  OUTPUT:
C   1.  THE DIRECTION E, OF THE SCREW AXIS, A 3-DIMENSIONAL VECTOR.
C   2.  THE LOCATION RHO OF THE POINT ON THE SCREW AXIS LYING CLOSEST
C       TO THE ORIGIN.
C   3.  THE ANGLE OF ROTATION (SIGN WITH RESPECT TO E
C       INCLUDED), THETA.
C   4.  THE SCALAR DISPLACEMENT DISPL, ALONG E (SIGN WITH RESPECT TO
C       E INCLUDED).
C
C  SUBSIDIARY SUBROUTINES:
C
C  CROSS(A,B,C)   COMPUTES THE CROSS PRODUCT OF VECTORS A AND B, IN THIS
C                 ORDER, AND STORES THE PRODUCT IN VECTOR C.
C  SCAL(A,B,S)    COMPUTES THE SCALAR PRODUCT OF VECTORS A AND B AND
C                 STORES THE PRODUCT IN THE SCALAR S.
C  EXCHGE(A,B)    EXCHANGES FIELDS A AND B.
C
C  3-DIMENSIONAL VECTORS A, B, C ARE AUXILIARY FIELDS.
C
      REAL AIN(3),BIN(3),CIN(3),AFIN(3),BFIN(3),CFIN(3),UA(3),UB(3)
     -    ,UC(3),A(3),B(3),C(3),E(3),RHO(3)
      LOGICAL LO(3)
      COMMON ZERO
C
C  COLLINEARITY OF GIVEN POINTS IS VERIFIED. WHEN POINTS ARE COLLINEAR,
C  ZERO IS SET EQUAL TO -1 AND SUBROUTINE RETURNS TO MAIN PROGRAM.
      DO 10 I=1,3
            LO(I)=.FALSE.
            A(I)=AIN(I)-CIN(I)
   10 B(I)=BIN(I)-CIN(I)
      CALL CROSS(A,B,C)
      CALL SCAL(C,C,S)
      S=SQRT(S)
      IF(S-ZERO) 20,20,30
   20 ZERO=-1.
      WRITE(6,1000)
      RETURN
C
C  DONE
C  COMPABILITY IS VERIFIED. IF THIS IS NOT MET, THEN ZERO IS SET EQUAL
C  TO -2, -3, OR -4, DEPENDING UPON WETHER DISTANCE AC, BC,  OR AB DOES
C  NOT REMAIN CONSTANT THROUGHOUT THE MOTION.
   30 DO 40 I=1,3
            C(I)=AFIN(I)-CFIN(I)
   40 CONTINUE
#
```

```
      CALL SCAL(A,A,S1)
      S1=SQRT(S1)
      CALL SCAL(C,C,S2)
      S2=SQRT(S2)
      IF(ABS(S1-S2).LE.ZERO) GO TO 50
      ZERO=-2.
      WRITE(6,1010)
      RETURN
   50 DO 60 I=1,3
              C(I)=BFIN(I)-CFIN(I)
   60 CONTINUE
      CALL SCAL(B,B,S1)
      CALL SCAL(C,C,S2)
      S1=SQRT(S1)
      S2=SQRT(S2)
      IF(ABS(S1-S2).LE.ZERO) GO TO 70
      ZERO=-3
      WRITE(6,1020)
      RETURN
   70 DO 80 I=1,3
              A(I)=AIN(I)-BIN(I)
   80 B(I)=AFIN(I)-BFIN(I)
      CALL SCAL(A,A,S1)
      CALL SCAL(B,B,S2)
      S1=SQRT(S1)
      S2=SQRT(S2)
      IF(ABS(S1-S2).LE.ZERO) GO TO 90
      ZERO=-4.
      WRITE(6,1030)
      RETURN
C
C  DONE
C  DISPLACEMENT VECTORS ARE COMPUTED
   90 DO 100 I=1,3
              UA(I)=AFIN(I)-AIN(I)
              UB(I)=BFIN(I)-BIN(I)
  100 UC(I)=CFIN(I)-CIN(I)
C
C  DONE
C  NUMBER OF VANISHING DISPLACEMENTS IS DETERMINED AND STORED IN NUZE.
C  DISPLACEMENT MAGNITUDES ARE STORED TEMPORARILY IN E. IF NUZE.EQ.0
C  THEN NUZE IS SET EQUAL TO 4.
      CALL SCAL(UA,UA,E(1))
      CALL SCAL(UB,UB,E(2))
      CALL SCAL(UC,UC,E(3))
      NUZE=0
      DO 110 I=1,3
              E(I)=SQRT(E(I))
              IF(E(I).GT.ZERO) GO TO 110
              NUZE=NUZE+1
              LO(I)=.TRUE.
  110 CONTINUE
```

```
C  DONE
      WRITE(6,1060) NUZE
      IF(NUZE.EQ.0)NUZE=4
C
C  EACH CASE (NUZE=1,2,3 AND 0) IS NOW INVESTIGATED.
      GO TO(111,211,311,411) NUZE
C
C  IF ONE, AND ONLY ONE, VECTOR VANISHES, THE SUBROUTINE INVESTIGATES
C  WHETHER THE REMAINING ONES ARE PARALLEL OR NOT. FIRST, THE
C  VANISHING VECTOR IS DETECTED BY SETTING THE INTEGER VARIABLE IN
C  EQUAL TO 1,2 OR 3, DEPENDING UPON WHETHER UA,UB OR UC, RESPECTIVELY,
C  IS THAT WHICH VANISHES.
  111 DO 120 I=1,3
              IF(LO(I))IN=I
  120 CONTINUE
      GO TO (121,131,141),IN
  121 CALL EXCHGE(UA,UC)
      CALL EXCHGE(AIN,CIN)
      CALL EXCHGE(AFIN,CFIN)
      GO TO 141
  131 CALL EXCHGE(UB,UC)
      CALL EXCHGE(BIN,CIN)
      CALL EXCHGE(BFIN,CFIN)
  141 CALL CROSS(UA,UB,C)
      CALL SCAL(C,C,S)
      S=SQRT(S)
C
C  THE FORMULA TO COMPUTE TH DIRECTION OF THE SCREW AXIS IS
C  CHOSEN DEPENDING UPON WHETHER THE TWO NONVANISHING VECTORS
C  ARE PARALLEL OR NOT.
      IF(S.GT.ZERO) GO TO 150
      WRITE(6,1070)
      GO TO 509
  150 WRITE(6,1080)
      GO TO 600
C
C  IF EXACTLY TWO VECTORS VANISH, IN IS SET EQUAL TO 1,2 OR 3, DEPEND-
C  ING UPON WHETHER UA, UB OR UC, RESPECTIVELY, IS THE NONVANISHING
C  VECTOR.
  211 DO 221 I=1,3
              IF(LO(I))GO TO 221
              IN=I
  221 CONTINUE
C
C  ALL VECTORS ARE RELABELLED SO THAT THE NONVANISHING VECTOR IS
C  STORED IN FIELD UC. VECTOR E IS THEN COMPUTED USING THE
C  FORMULA REALISED IN STATEMENTS 501-520.
      GO TO (231,232,233),IN
  231 CALL EXCHGE(AIN,CIN)
      CALL EXCHGE(AFIN,CFIN)
      CALL EXCHGE(UA,UC)
      GO TO 233
  232 CALL EXCHGE(BIN,CIN)
      CALL EXCHGE(BFIN,CFIN)
      CALL EXCHGE(UB,UC)
```

```
C  UA-UC AND UB-UC ARE STORED IN UA AND UB, RESPECTIVELY.
  233 DO 240 I=1,3
             UA(I)=UA(I)-UC(I)
  240 UB(I)=UB(I)-UC(I)
      GO TO 509
C
C  IF ALL THREE VECTORS VANISH,A MESSAGE IS WRITEN AND SCREW RE-
C  TURNS TO MAIN PROGRAM.
  311 WRITE(6,1040)
      ZERO=-5
      RETURN
C
C  DONE
C  ONE-, TWO- AND THREE- VANISHING-DISPLACEMENT CASES WERE ALREADY
C  DEALT WITH. NO-VANISHING-DISPLACEMENT CASE IS NEXT INVESTIGATED.
C  FIRST IT IS INVESTIGATED WHETHER THE GIVEN VECTORS ARE
C  COPLANAR AND, IF SO, WHETHER THEIR DIFFERENCES ARE PARALLEL.
C  THE FORMULA EMPLOYED TO COMPUTE VECTOR E IS DECIDED UPON
C  ACCORDINGLY.
  411 CALL CROSS(UA,UB,C)
      CALL SCAL(UC,C,S1)
C
C  UA-UC AND UB-UC ARE STORED IN UA AND UB RESPECTIVELY.
      DO 420 I=1,3
             UA(I)=UA(I)-UC(I)
  420 UB(I)=UB(I)-UC(I)
      CALL CROSS (UA,UB,C)
      CALL SCAL(C,C,S2)
      S2=SQRT(S2)
      JUMP=1
      IF(ABS(S1).LT.ZERO)JUMP=JUMP+1
      IF(S2.LT.ZERO)JUMP=JUMP+1
      GO TO (421,422,423),JUMP
  421 WRITE(6,1090)
      GO TO 600
  422 WRITE(6,1100)
      GO TO 600
C
C  IF VECTORS ARE COPLANAR AND DIFFERENCES ARE PARALLEL, INVESTIGATES
C  WHETHER MOTION IS A PURE TRANSLATION.
  423 CALL SCAL(UA,UA,UAUA)
      CALL SCAL(UB,UB,UBUB)
      UAUA=SQRT(UAUA)
      UBUB=SQRT(UBUB)
      IF(UAUA.LE.ZERO.AND.UBUB.LE.ZERO)GO TO 900
      WRITE(6,1110)
C
C  COMPUTES VECTOR E. FIRST THE SCALAR FACTOR BETA RELATING
C  PARALLEL VECTORS UA AND UB IS COMPUTED-VECTOR E IS STORED
C  TEMPORARILY IN C.
  509 CALL SCAL(UA,UA,UAMAG)
      CALL SCAL(UB,UB,UBMAG)
      CALL SCAL(UA,UB,UAUB)
```

```
      UAMAG=SQRT(UAMAG)
      UBMAG=SQRT(UBMAG)
      IF(UAMAG.LE.ZERO) WRITE(6,1120)
      BETA=UBMAG/UAMAG
      IF(UAUB.LE.0)BETA=-BETA
      DO 510 I=1,3
  510 E(I)=BIN(I)-CIN(I)-BETA*(AIN(I)-CIN(I))
      GO TO 605
  600 CALL CROSS(UA,UB,E)
  605 CALL SCAL(E,E,EE)
      EE=SQRT(EE)
      DO 610 I=1,3
  610 E(I)=E(I)/EE
C
C   DONE
C   THE ANGLE OF ROTATION IS COMPUTED.AUXILIARY VECTORS A AND B
C   ARE INTRODUCED.
      DO 710 I=1,3
             A(I)=AIN(I)-CIN(I)
             C(I)=AIN(I)+UA(I)
  710 B(I)=C(I)-CIN(I)
      CALL SCAL(A,E,APROY)
      CALL SCAL(B,E,BPROY)
      DO 720 I=1,3
             A(I)=A(I)-APROY*E(I)
  720 B(I)=B(I)-BPROY*E(I)
      CALL SCAL(A,B,AB)
      CALL SCAL(A,A,AA)
      THETA=ARCOS(AB/AA)
      CALL CROSS(A,B,C)
      CALL SCAL(C,E,CE)
      IF(CE.LE.0)THETA=-THETA
C
C  DONE
C   COMPUTATION OF RHO AND DISPL.
      COTH=COS(THETA)
      DIF=1+COSH
      DIF=ABS(DIF)
      IF(DIF.LE.ZERO) GO TO 830
      DO 810 I=1,3
             UA(I)=AFIN(I)-AIN(I)
  810 C(I)=AFIN(I)+AIN(I)
      CALL CROSS(E,C,A)
      CALL CROSS(E,A,B)
      CALL CROSS(E,UA,RHO)
      DO 820 I=1,3
             RHO(I)=COTAN(THETA/2.)*RHO(I)
  820 RHO(I)=(RHO(I)-B(I))/2.
      GO TO 840
  830 DO 850 I=1,3
  850 RHO(I)=AIN(I)+UA(I)*0.5
      CALL SCAL(RHO,E,S)
      DO 860 I=1,3
  860 RHO(I)=RHO(I)-S*E(I)
```

```
 840 CALL SCAL(UA,E,DISPL)
     RETURN
 900 DO 905 I=1,3
 905 UA(I)=AFIN(I)-AIN(I)
     WRITE(6,1050)(UA(I),I=1,3)
     ZERO=-6
     RETURN
1000 FORMAT(5X,'POINTS ARE COLLINEAR.MOTION IS UNDEFINED'/)
1010 FORMAT(5X,'MOTION IS NOT RIGID.LENGTH AC DOES NOT REMAIN',
    -       ' CONSTANT.'/)
1020 FORMAT(5X,'MOTION IS NOT RIGID.LENGTH BC DOES NOT REMAIN',
    -       ' CONSTANT.'/)
1030 FORMAT(5X,'MOTION IS NOT RIGID.LENGTH AB DOES NOT REMAIN',
    -       ' CONSTANT.'/)
1040 FORMAT(5X,'NO MOTION.ALL THREE DISPLACEMENTS VECTORS ARE',
    -       ' ZERO.'/)
1050 FORMAT(9X,'VECTORS DIFFERENCES ARE BOTH ZERO '/
    -       9X,'THE MOTION IS PURE TRANSLATION. '//15X,'THE ',
    -       'DISPLACEMENT HAS THE FOLLOWING X-,Y-AND Z COMPONENTS :'
    -       /15X,F12.5,5X,F12.5,5X,F12.5,/)
1060 FORMAT(9X,'NUMBER OF VANISHING DISPLACEMENTS IS',I5)
1070 FORMAT(9X,'DIFFERENCES ARE PARALLEL'/)
1080 FORMAT(9X,'DIFFERENCES ARE NONPARALLEL'/)
1090 FORMAT(9X,'VECTORS ARE NONCOPLANAR'/)
1100 FORMAT(9X,'VECTORS ARE COPLANAR AND DIFFERENCES ARE NONPARALLEL'/)
1110 FORMAT(9X,'VECTOR DIFFERENCES ARE PARALLEL'/)
1120 FORMAT(1X,'THE MAGNITUD OF THE NONVANISING VECTOR UA WAS '
    -      'COMPUTED AS ZERO BETWEEN STATEMENS 509 AND 510',/)
     END
```

```
          $
C   THIS PROGRAM COMPUTES THE PARAMETERS OF THE INSTANTANEOUS
C   SCREW OF GIVEN RIGID-BODY MOTIONS.
C
      REAL A(3),B(3),C(3),VA(3),VB(3),VC(3),RHO(3),SPIN(3)
      COMMON ZERO
      NUM=0
1     READ(5,10,END=50)N
      ZERO=0.00001
      NUM=NUM+1
      DO 2 I=1,N
           READ(5,20)A(I),B(I),C(I)
           READ(5,20)VA(I),VB(I),VC(I)
2     CONTINUE
      WRITE(6,30)NUM
      WRITE(6,31)
      WRITE(6,32)(A(I),I=1,N)
      WRITE(6,33)(B(I),I=1,N)
      WRITE(6,34)(C(I),I=1,N)
      WRITE(6,35)(VA(I),I=1,N)
      WRITE(6,36)(VB(I),I=1,N)
      WRITE(6,37)(VC(I),I=1,N)
      CALL INSCRU(A,B,C,VA,VB,VC,RHO,SPIN,SPEED)
      IF(ZERO.LT.0.0) GO TO 1
      WRITE(6,38)(SPIN(I),I=1,N)
      WRITE(6,39)(RHO(I),I=1,N)
      WRITE(6,40)SPEED
      GO TO 1
10    FORMAT(I2)
20    FORMAT(F7.3)
30    FORMAT(5X,'EXAMPLE',I3/)
31    FORMAT(5X,'POINT',20X,'COORDINATES'/18X,'X',16X,'Y',16X,'Z'/)
32    FORMAT(5X,'A  = ',F12.5,5X,F12.5,5X,F12.5)
33    FORMAT(5X,'B  = ',F12.5,5X,F12.5,5X,F12.5)
34    FORMAT(5X,'C  = ',F12.5,5X,F12.5,5X,F12.5)
35    FORMAT(5X,'VA = ',F12.5,5X,F12.5,5X,F12.5)
36    FORMAT(5X,'VB = ',F12.5,5X,F12.5,5X,F12.5)
37    FORMAT(5X,'VC = ',F12.5,5X,F12.5,5X,F12.5)
38    FORMAT(9X,'THE SPIN HAS THE FOLLOWING COMPONENTS'/
     -       9X,'(VECTOR SPIN) :',F12.5,5X,F12.5,5X,F12.5)
39    FORMAT(9X,'THE POINT ON THE INSTANTANEOUS-SCREW AXIS CLOSEST'
     -       ,' TO'/9X,'THE ORIGIN HAS THE FOLLOWING X-, Y- AND Z',
     -       ' COORDINATES'/9X,'(VECTOR RHO) :',F12.5,5X,F12.5,5X,
     -       F12.5)
40    FORMAT(9X,'THE VELOCITY COMPONENT ALONG THE INSTANTANEOUS-SCREW',
     -       ' AXIS :'/9X,'(SPEED) :',F12.5)
50    CALL EXIT
      END
*
```

```
      SUBROUTINE INSCRU(A,B,C,VA,VB,VC,RHO,SPIN,SPEED)
C
C  THIS SUBROUTINE COMPUTES THE PARAMETERS OF THE INSTANTANEOUS
C  SCREW OF A RIGID-BODY MOTION.
C  INPUT:
C  THE X,Y AND Z COORDINATES OF THREE NONCOLINEAR POINTS,
C  A,B AND C OF A RIGID BODY AND THEIR CORRESPONDING
C  VELOCITIES, VA,VB AND VC. FIELDS A,B,C,VA,VB AND VC ARE
C  THREE DIMENSIONAL, THEIR ELEMENTS BEING THE X,Y AND Z
C  COMPONENTS OF THE CARTESIAN VECTORS STORED IN THEM.
C  OUTPUT:
C  1.- THE POSITION VECTOR RHO OF THE POINT ON THE INSTANTANEOUS
C      -SCREW AXIS LYING CLOSEST TO THE ORIGIN.
C  2.- THE SPIN. THIS VARIABLE IS LABELED 'SPIN'.
C  3.- THE SLIDING SPEED, I.E THE PROJECTION OF THE VELOCITY FIELD
C      ON THE AXIS OF THE INSTANTANEOUS SCREW. THIS VARIABLE IS
C      LABELED 'SPEED'.
C  SUBSIDIARY SUBROUTINES:
C  EXCHGE(A,B) -EXCHANGES FIELDS A AND B.
C  CROSS(U,V,W)-COMPUTES THE CROSS PRODUCT  OF VECTOR U TIMES
C              VECTOR V, IN THIS ORDER, AND STORES THE PRODUCT
C              IN VECTOR W.
C  SCAL(U,V,S) -COMPUTES THE SCALAR PRODUCT OF VECTORS U AND
C              V AND STORES THE PRODUCT IN THE SCALAR S.
C
      REAL A(3),B(3),C(3),VA(3),VB(3),VC(3),RHO(3),SPIN(3),AU(3),BU(3),
     -     CU(3)
      LOGICAL LO(3)
      COMMON ZERO
C
C  IT IS VERIFIED WHETHER THE GIVEN POINTS ARE COLLINEAR. IF
C  THEY ARE, ZERO IS SET EQUAL TO -1 AND SUBROUTINE RETURNS
C  TO MAIN PROGRAM. DIFFERENCE VECTORS ARE TEMPORARILY STORED
C  IN THE SPIN AND RHO FIELDS.
C
      DO 10 I=1,3
         SPIN(I)=A(I)-C(I)
10    RHO(I)=B(I)-C(I)
      CALL CROSS(SPIN,RHO,CU)
      CALL SCAL(CU,CU,S)
      S=SQRT(S)
      IF(S.GT.ZERO) GO TO 30
      ZERO=-1
      WRITE(6,1000)
      RETURN
C
C  DONE
C  COMPATIBILITY IS VERIFIED. IF THIS IS NOT MET, ZERO IS
C  SET EQUAL TO -2.,-3.,-4., DEPENDING UPON WHETHER DISTANCE
C  AC, BC OR AB HAS A NONZERO TIME RATE OF CHANGE.
C
```

```
         $
30      CALL SCAL(VA,SPIN,SA)
        CALL SCAL(VC,SPIN,SC)
        IF(ABS(SA-SC).LE.ZERO) GO TO 40
        ZERO=-2.
        WRITE(6,1010)
        RETURN
40      CALL SCAL(VB,RHO,SB)
        CALL SCAL(VC,RHO,SC)
        IF(ABS(SB-SC).LE.ZERO) GO TO 50
        ZERO=-3.
        WRITE(6,1020)
        RETURN
50      DO 60 I=1,3
60      AU(I)=B(I)-A(I)
        CALL SCAL(VA,AU,SA)
        CALL SCAL(VB,AU,SB)
        IF(ABS(SA-SB).LE.ZERO) GO TO 70
        ZERO=-4.
        WRITE(6,1030)
        RETURN
C
C  DONE
C  NUMBER OF VANISHING-VELOCITY VECTORS IS DETERMINED AND
C  STORED IN NUZE. DISPLACEMENT MAGNITUDES ARE TEMPORARILY
C  STORED IN AU. IF NUZE.EQ.0, THEN NUZE IS SET EQUAL TO 4.
C
70      CALL SCAL(VA,VA,AU(1))
        CALL SCAL(VB,VB,AU(2))
        CALL SCAL(VC,VC,AU(3))
        NUZE=0
        DO 80 I=1,3
              LO(I)=.FALSE.
              AU(I)=SQRT(AU(I))
              IF(AU(I).GT.ZERO) GO TO 80
              NUZE=NUZE+1
              LO(I)=.TRUE.
80      CONTINUE
        WRITE(6,1040)NUZE
        IF(NUZE.EQ.0) NUZE=4
C
C  DONE
C  EACH CASE (NUZE=1,2,3,0) IS NOW INVESTIGATED. PARALLELISM AND
C  EQUALITY OF VECTORS ARE THEN VERIFIED.
C
        GO TO(100,200,300,400),NUZE
C
C  IF ONE, AND ONLY ONE VELOCITY VECTOR VANISHES, THE
C  SUBROUTINE INVESTIGATES WHETHER THE REMAINING ONES ARE
C  PARALLEL OR NOT.
C
*
```

```
100     DO 101 I=1,3
                IF(LO(I)) IN=I
101     CONTINUE
        GO TO (110,120,130),IN
110     CALL EXCHGE(VA,VC)
        CALL EXCHGE(A,C)
        GO TO 130
120     CALL EXCHGE(VB,VC)
        CALL EXCHGE(B,C)
130     CALL CROSS(VA,VB,AU)
        CALL SCAL(AU,AU,S)
        S=SQRT(S)
        DO 140 I=1,3
                AU(I)=VA(I)
                BU(I)=VB(I)
140     RHO(I)=B(I)-C(I)
C
C  THE FORMULA TO COMPUTE THE DIRECTION OF THE SCREW AXIS IS
C  CHOSEN DEPENDING UPON WHETHER THE TWO NONVANISHING VECTORS
C  ARE PARALLEL OR NOT.
C
        IF(S.GT.ZERO) GO TO 150
        WRITE(6,1050)
        GO TO 500
150     WRITE(6,1060)
        GO TO 600
C
C  IF EXACTLY TWO VECTORS VANISH, IN IS SET EQUAL TO 1,2 OR 3
C  DEPENDING UPON WHETHER VA,VB OR VC, RESPECTIVELY, IS THE
C  NONVANISHING VECTOR.
C
200     DO 202 I=1,3
                IF(LO(I)) GO TO 202
                IN=I
202     CONTINUE
C
C  ALL VECTORS ARE RELABELLED SO THAT THE NONVANISHING
C  VELOCITY VECTOR IS STORED IN VC.
C
        GO TO(210,220,230),IN
210     CALL EXCHGE(A,C)
        CALL EXCHGE(VA,VC)
        GO TO 230
220     CALL EXCHGE(B,C)
        CALL EXCHGE(VB,VC)
C
C  VA-VC AND VB-VC ARE STORED IN AU AND BU RESPECTIVELY.
C
230     DO 240 I=1,3
                AU(I)=VA(I)-VC(I)
```

```
240     BU(I)=VB(I)-VC(I)
        GO TO 500
C
C  IF ALL THREE VELOCITIES VANISH, A MESSAGE IS WRITTEN AND INSCRU
C  RETURNS TO THE MAIN PROGRAM.
C
300     WRITE(6,1070)
        ZERO=-5
        RETURN
C
C  DONE
C  ONE, TWO AND THREE-VANISHING-VELOCITY CASES WERE ALREADY DEALT
C  WITH. NO-VANISHING DISPLACEMENT CASE IS NOW INVESTIGATED. FIRST
C  IT IS INVESTIGATED WHETHER THE GIVEN VECTORS ARE COPLANAR AND
C  IF SO, WHETHER THEIR DIFFERENCES ARE PARALLEL.
C
400     CALL CROSS(VA,VB,CU)
        CALL SCAL(VC,CU,S1)
C
C  VA-VC AND VB-VC ARE STORED IN AU AND BU RESPECTIVELY.
C
        DO 402 I=1,3
               AU(I)=VA(I)-VC(I)
402     BU(I)=VB(I)-VC(I)
        CALL CROSS(AU,BU,CU)
        CALL SCAL(CU,CU,S2)
        S2=SQRT(S2)
        IND=1
        IF(ABS(S1).LT.ZERO)IND=IND+1
        IF(S2.LT.ZERO)IND=IND+1
        GO TO (410,420,430),IND
410     WRITE(6,1080)
        GO TO 600
420     WRITE(6,1090)
        GO TO 600
C
C  IF VECTORS ARE COPLANAR AND DIFERENCES ARE PARALLEL, INVESTIGATES
C  WHETHER THE MOTION IS A PURE TRANSLATION.
C
430     CALL SCAL(AU,AU,VA2)
        CALL SCAL(BU,BU,VB2)
        VA2=SQRT(VA2)
        VB2=SQRT(VB2)
        IF(VA2.LE.ZERO.AND.VB2.LE.ZERO) GO TO 700
        WRITE(6,1100)
C
C  COMPUTES VECTOR SPIN. FIRST THE SCALAR FACTOR ALFA IS COMPUTED.
C
500     DO 502 I=1,3
               RHO(I)=B(I)-C(I)
```

```
502       SPIN(I)=A(I)-C(I)
          CALL SCAL(AU,AU,VA2)
          CALL CROSS(RHO,SPIN,CU)
          CALL SCAL(CU,CU,ABC)
          VA2=SQRT(VA2)
          ABC=SQRT(ABC)
          ALFA=VA2/ABC
          CALL SCAL(AU,CU,SG)
          IF(SG.LT.0)ALFA=-ALFA
C
C   DONE
C   THE SCALAR FACTOR BETA IS COMPUTED
C
          CALL SCAL(BU,BU,SB)
          CALL SCAL(AU,AU,SA)
          CALL SCAL(AU,BU,SAB)
          SB=SQRT(SB)
          SA=SQRT(SA)
          BETA=SB/SA
          IF(SAB.LT.0)BETA=-BETA
C
C   DONE
C
          DO 504 I=1,3
                 SPIN(I)=RHO(I)-BETA*SPIN(I)
504       SPIN(I)=SPIN(I)*ALFA
C
C   DONE
C   RHO AND SPIN ARE NEXT COMPUTED
C
          GO TO 610
600       CALL CROSS(AU,BU,CU)
          CALL SCAL(AU,RHO,DEN)
          DO 602 I=1,3
602       SPIN(I)=CU(I)/DEN
610       CALL SCAL(SPIN,SPIN,W2)
          CALL SCAL(SPIN,A,COFW)
          CALL CROSS(SPIN,VA,RHO)
          DO 612 I=1,3
612       RHO(I)=A(I)+(RHO(I)-COFW*SPIN(I))/W2
          CALL SCAL(VA,SPIN,SPEED)
          W2=SQRT(W2)
          SPEED=SPEED/W2
C
C   DONE
C
          RETURN
700       WRITE(6,1110)(VA(I),I=1,3)
          ZERO=-6
          RETURN
```

```
      $
1000     FORMAT(5X,'POINTS ARE COLINEAR. MOTION IS UNDEFINED'/)
1010     FORMAT(5X,'MOTION IS NOT RIGID. LENGTH AC DOES NOT REMAIN',
     -            ' CONSTANT'/)
1020     FORMAT(5X,'MOTION IS NOT RIGID. LENGTH BC DOES NOT REMAIN',
     -            ' CONSTANT'/)
1030     FORMAT(5X,'MOTION IS NOT RIGID. LENGTH AB DOES NOT REMAIN',
     -            ' CONSTANT'/)
1040     FORMAT(5X,'NUMBER OF VANISHING VELOCITIES IS :',I5/)
1050     FORMAT(5X,'VELOCITIES ARE PARALLEL'/)
1060     FORMAT(5X,'VELOCITIES ARE NONPARALLEL'/)
1070     FORMAT(5X,'NO MOTION. ALL THREE VELOCITY VECTORS ARE ZERO'/)
1080     FORMAT(5X,'VELOCITIES ARE NONCOPLANAR'/)
1090     FORMAT(5X,'VELOCITIES ARE COPLANAR AND DIFFERENCES ARE NON',
     -            'PARALLEL'/)
1100     FORMAT(5X,'VELOCITIES ARE COPLANAR AND DIFERENCES ARE ',
     -            'PARALLEL'/)
1110     FORMAT(5X,'MOTION IS PURE TRANSLATION'/5X,'THE VELOCITIES ',
     -            'HAVE THE FOLLOWING X-,Y- AND Z- COORDINATES:'/
     -             3(5X,F12.5)/)
         END
*
```

```
      $
         SUBROUTINE EXCHGE(A,B)
C
C  THIS SUBROUTINE EXCHANGES THE FIELDS A AND B, I.E. IT RETURNS
C  B AS A AND A AS B.
C
C
         REAL A(3),B(3),AUX(3)
         DO 10 I=1,3
               AUX(I)=A(I)
               A(I)=B(I)
               B(I)=AUX(I)
10       CONTINUE
         RETURN
         END
*
```

```
      SUBROUTINE CROSS(A,B,C)
C
C  THIS SUBROUTINE PERFORMS THE CROSS PRODUCT A AND B,IN THIS ORDER,
C  AND STORES THIS PRODUCT IN C.
C
C
      REAL A(3),B(3),C(3)
      DO 10 K=1,3
            C(K)=0.
            DO 10 L=1,3
                  DO 10 M=1,3
                        N=(L-K)*(M-L)*(K-M)
                        C(K)=C(K)-N*A(L)*B(M)/2.
10    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SCAL(A,B,S)
C
C  THIS SUROUTINE PERFORMS THE SCALAR PRODUCT OF VECTORS A AND B
C  AND STORES THIS PRODUCT IN S.
C
C
      REAL A(3),B(3)
      S=0.
      DO 10 I=1,3
            S=S+A(I)*B(I)
10    CONTINUE
      RETURN
      END
```

REFERENCES

1. Angeles J., "On the computation of the screw parameters of a rigid body motion. Part I: Finitely-separated positions", Internal Report, DEPFI-UNAM (Graduate Division of the Faculty of Engineering, National University of Mexico), Mexico City, 1981.

2. Angeles J., "On the computation of the screw parameters of a rigid body motion. Part II: Infinitesimally-separated positions", Internal Report, DEPFI-UNAM (Graduate Division of the Faculty of Engineering, National University of Mexico), Mexico City, 1981.