Policy on the Redistribution of DTIC-Supplied Information

As a condition for obtaining DTIC services, all information received from DTIC that is not clearly marked for public release will be used only to bid or perform work under a U.S. Government contract or grant or for purposes specifically authorized by the U.S. Government agency that is sponsoring access. Further, the information will not be published for profit or in any manner offered for sale. Non-compliance may result in termination of access and a requirement to return all information obtained from DTIC.

NOTICE

We are pleased to supply this document in response to your request.

The acquisition of technical reports, notes, memorandums, etc., is an active, ongoing program at the Defense Technical Information Center (DTIC) that depends, in part, on the efforts and interest of users and contributors.

Therefore, if you know of the existence of any significant reports, etc., that are not in the DTIC collection, we would appreciate receiving copies or information related to their sources and availability.


Our Programs Management Branch, DTIC-OCP, will assist in resolving any questions you may have concerning documents to be submitted. Telephone numbers for that office are (703) 274-6847, or DSN 284-6847. The Reference Services Branch, DTIC-BCR, will assist in document identification, ordering and related questions. Telephone numbers for that office are (703) 274-7633 or DSN 284-7633.

DO NOT RETURN THIS DOCUMENT TO DTIC

EACH ACTIVITY IS RESPONSIBLE FOR DESTRUCTION OF THIS DOCUMENT ACCORDING TO APPLICABLE REGULATIONS.
MINIMAL-TIME SHIP ROUTING

by

W. E. BLEICK and F. D. FAULKNER

Professors of Mathematics and Mechanics

COPY OF

HARD COPY $ .20
MICROFICHE $ .05

Research Paper No. 46

UNITED STATES NAVAL POSTGRADUATE SCHOOL
Monterey, California
August 1964
Minimal-Time Ship Routing
V. E. BLEICK and P. D. FAULKNER
U. S. Naval Postgraduate School, Monterey, Calif.

ABSTRACT
A recent theory of minimal-time ship routing through time-dependent ocean wave height and direction fields is put to a numerical test by using a series of semidaily analyses furnished by the U. S. Navy Fleet Numerical Weather Facility. The interpolations and integrations required are found to be feasible. A resume of the theory is given.

1. Introduction
Haltiner, Hamilton and 'Arnason (1962) gave a relaxation method solution to the problem of minimal-time routing of ships through ocean wave height and direction fields dependent on the ship location coordinates only. The theory has been extended by Faulkner (1963) to the case where the wave height and direction depend on time also. The present paper confronts this time-dependent theory with actual wave height and direction analyses from the files of the U. S. Navy Fleet Numerical Weather Facility, and reports on practical problems which had to be solved in a test of the theory.

2. Polar velocity diagram
A basic ingredient of the theory is the polar diagram of Fig. 1, giving the ship velocity \( v \) as a function of the angle \( \theta \) between the ship's heading and the wave direction. A diagram of this kind must be specified for each wave height \( H \). The points L, M and N on the diagram correspond to the ship speed \( v_h \) in head waves, \( v_b \) in beam waves, and \( v_f \) in following waves. Empirical curves for these three speeds as functions of wave height \( H \) are available in the pioneer
Fig. 1. Polar
diagram.
work of James (1957). His P2-S2-32 ship type curves, shown in Fig. 2, have been chosen for use here. They have the appearance of being arcs of hyperbolas, at least approximately. This was confirmed when a least squares analysis showed that all three of \( v_h, v_d \) and \( v_f \) can be represented closely as functions of \( H \) by the hyperbolic arc

\[
c_1^2 + c_2^2 H^2 - (c_3 + c_4 H)^2 = (c_1 - c_0)^2 - c_3^2 c_4^2 \tag{1}
\]

where \( c_0 \) is the point common to all three speeds when \( H = 0 \). The other four constants are related to the asymptotes \( (c_2 + c_4)H + c_1 - c_3 \) and \( (c_2 - c_4)H + c_1 - c_3 \). It was decided to construct the polar velocity diagram by fitting an ellipse to the points \( L, M \) and \( N \). This resulted in semi-principal axes \( a = (v_h + v_f)/2 \), \( b = \sqrt{v_b/(v_h v_f)} \), and a distance to the eccentric pole \( O \) given by \( c = (v_f - v_h)/2 \). Note that the pole \( O \) is not to be construed as a focus of the ellipse. A further least squares analysis showed that the semi-axes \( a \) and \( b \) are closely representable also by hyperbolic functions of the form of (1), but not \( c \) which must be calculated as \( c = (v_f - v_h)/2 \).

3. Coordinate system

The ocean wave height and direction data of the semidaily Fleet Numerical Weather Facility analyses are presented in a south-polar stereographic projection of the northern hemisphere upon a plane passing through the circle of \( 60^\circ \) North latitude. A rectangular coordinate system is set up in this projecting plane with the \( O_x \) and \( O_y \) axes parallel to the projections of the meridians of \( 10^\circ \) and \( 100^\circ \) East longitude respectively. A 62 by 62 grid is constructed using these axes with \( x = y = 31 \) defining the projection of the North pole. The mesh distance between grid lines corresponds to a distance of 381 km at \( 60^\circ \) North latitude where the projection is true. The radius of the equator's projection is 31,205 mesh units. The map scale factor \( m \),
Fig. 2. Ship speed in "JUNE 1959 AFTER JAMES P2-S2-R2 SHIP TYPES"
; beam and head waves.
defined as the ratio of a differential distance in the Oxy plane to the corresponding differential distance on the earth's surface, is

$$m = \frac{[973.75 + (x-31)^2 + (y-31)^2]}{1043.6}$$  \hspace{1cm} (2)

Let \( x, y \) be the coordinates of a ship's projection in the Oxy plane at time \( t \). Then the projected speed of the ship is

$$V(x, y, t, \hat{\varepsilon}) = v(H, \hat{\varepsilon}) m$$  \hspace{1cm} (3)

where \( v(H, \hat{\varepsilon}) \) is the actual geographical ship speed of the polar velocity diagram of Fig. 1, and where \( H(x, y, t) \) is obtained by interpolation in the Fleet Numerical Weather Facility grid wave height data. Since the stereographic projection is a conformal transformation preserving angles and their senses, the angle \( \hat{\varepsilon} \) is the same in the Oxy plane as on the earth's surface.

4. Resume of the theory

Fig. 3 shows a ship at the point \((x, y, t)\) in the stereographic projection plane on a route from fixed initial point \( A \) at \( t=0 \) to fixed terminal point \( B \) at \( t=T \). The elliptical polar velocity diagram for \( V=mv \) is plotted at this point by interpolation in the semidaily wave height \( H(x, y, t) \) and wave direction \( K(x, y, t) \) grid values of the Fleet Numerical Weather Facility. The special type of interpolation required is discussed in the Appendix. The direction of the ship's velocity vector \( \mathbf{V} = i\ddot{x} + j\ddot{y} \) is the control angle \( p \), which is to be chosen at each point so as to minimize the transit time \( T \) from \( A \) to \( B \). The equations of motion of the ship's projection in the Oxy plane are

$$\varphi_1 = \ddot{x} - V \cos p = 0, \quad \varphi_2 = \ddot{y} - V \sin p = 0$$  \hspace{1cm} (4)

where \( V=|\mathbf{V}| = V(x, y, t, p) \). The problem of minimizing \( T \) is equivalent to the Lagrange calculus of variations problem of requiring the integral

$$I = \int_0^T (1 + \lambda \varphi_1 + \mu \varphi_2) dt$$  \hspace{1cm} (5)
BLANK PAGE
Fig. 3. Ship motion
A \gamma = 0
cographic Oxy plane.
to be stationary, where \( \lambda(t) \) and \( \mu(t) \) are continua of Lagrangian multipliers. Let the time at the fixed terminal point \( B \) be varied to \( T+\Delta T \). The vanishing first variation of \( I \) is

\[
\delta I = \Delta T + [\delta \lambda x + \mu \delta y]_0^T - \int_0^T (\dot{\varphi}_3 \delta x + \dot{\varphi}_4 \delta y + \dot{\varphi}_3 \delta p) \, dt = 0. \tag{6}
\]

The coefficients of \( \delta x, \delta y, \delta p \) in \( \delta I = 0 \) give the Euler equations (7), (8) and (9), consisting of the adjoint equations

\[
\begin{align*}
\dot{\varphi}_3 &= \dot{\lambda} + (\lambda \cos \mu \sin \nu) V_x = 0, \tag{7} \\
\dot{\varphi}_4 &= \dot{\mu} + (\lambda \cos \mu \sin \nu) V_y = 0. \tag{8}
\end{align*}
\]

and the scalar product control equation

\[
\varphi_5 = \overline{\lambda} \cdot \overline{V}_p = 0, \tag{9}
\]

where the adjoint vector \( \overline{\lambda} = \lambda + j \mu \), and where \( V = V_\lambda \delta \lambda + \frac{V_\mu}{\lambda} \delta \mu \) and \( V_\lambda, V_\mu \) are the tangent vector to the polar velocity diagram of Fig.3. Eq. (9) implies the orthogonality of \( \overline{\lambda} \) and \( \overline{V}_p \) as shown in Fig.3. Eq. (9) may be written also in the form

\[
p = \arctan(\mu/\nu) + \arctan(V_\lambda/V_\mu). \tag{10}
\]

The fixed endpoints \( A \) and \( B \) imply that

\[
\begin{align*}
\delta x(0) &= 0, \quad \delta y(0) = 0, \tag{11} \\
\delta x(T) &= (\dot{x} \Delta t + \delta x)_T = 0, \quad \delta y(T) = (\dot{y} \Delta t + \delta y)_T = 0. \tag{12}
\end{align*}
\]

Use of Eqs. (11) and (12) makes the remaining terms of (6) proportional to \( \Delta T \), whose coefficient gives the scalar product transversality condition

\[
(\overline{\lambda} \cdot \overline{V})_T = 1 > 0 \tag{13}
\]

meaningful for sign only because of the homogeneity of (7) and (8). Eq. (13) implies that the angle \( \angle \) between \( \overline{V} \) and \( \overline{\lambda} \) is acute, as shown in Fig.3. A further implication of (10) and (13) is that the quadrant of \( \nu \) is such that

\[
\cos \nu = (\lambda V - \mu V_\lambda)/\overline{\lambda} R, \quad \sin \nu = (\lambda V + \mu V_\mu)/\overline{\lambda} R, \tag{14}
\]
where $\Lambda = |\Lambda| = (\lambda^2 + \omega^2)^2$ and $R = |R_p| = (v^2 + v_p^2)^2$.

The simultaneous numerical integration of (4), (7), (8) and (14) is carried out together with a Newton-Raphson iteration as follows:

Let $\lambda_1, \mu_1$ and $\lambda_2, \mu_2$ be two linearly independent solutions of the adjoint Eqs. (7) and (8) corresponding to the columns of the matrix

$$E(t) = \begin{bmatrix} \lambda_1 & \lambda_2 \\ \mu_1 & \mu_2 \end{bmatrix}$$

(15)

where $E(0) = I$ is the identity matrix. The $\lambda, \mu$ of (14) are taken as the linear combinations $\lambda = \lambda_1 \cos \alpha + \lambda_2 \sin \alpha$ and $\mu = \mu_1 \cos \alpha + \mu_2 \sin \alpha$. The variation $\delta p$ is found by total differentiation of (10) to be

$$\delta p = R |E| \delta \alpha / \Lambda^2 (v^2 + 2v_p^2 - vv_p^2)$$

(16)

where $|E|$ is the determinant of $E$. Assume that a solution of the ship motion Eqs. (4) has been found, corresponding to (7), (8) and (14) for some value of $\alpha$, which falls short of the fixed end point $B$ at $t = T$ by the coordinate differences $\Delta x(T)$ and $\Delta y(T)$. Using this solution and holding $T$ fixed, find the variation of the vanishing matrix integral

$$\int_0^T [\Phi_1, \Phi_2] E(t) dt = 0.$$  

(17)

Since the columns of $E(t)$ satisfy the adjoint Eqs. (7) and (8), one obtains the $1 \times 2$ matrix equation

$$[\delta x, \delta y]_T E(T) = \int_0^T [(v_p \cos \alpha - v \sin \alpha), (v_p \sin \alpha + v \cos \alpha)] E \delta \alpha dt.$$

(18)

Substitution from (14) and (16) into (18) gives

$$[\delta x, \delta y]_T = [-\mu, \lambda]_T J \delta \alpha$$

(19)

where

$$J = \frac{1}{|E(T)|} \int_0^T R^3 |E|^2 dt / \Lambda^3 (v^2 + 2v_p^2 - vv_p^2).$$

(20)

Now vary the terminal time from $T$ to $T + \Delta T$ and substitute

$$[\delta x, \delta y]_T = [\Delta x, \Delta y]_T - [\dot{x}, \dot{y}]_T \Delta T$$

(21)
into (19) to obtain the Newton-Raphson equations

\[
\begin{align*}
\dot{x}(T)\Delta T - J_u(T)\delta \alpha &= \Delta x(T) \\
\dot{y}(T)\Delta T + J_\lambda(T)\delta \alpha &= \Delta y(T)
\end{align*}
\]

(22)

for the determination of \( \Delta T \) and \( \delta \alpha \) on a varied trajectory which attempts to correct the errors \( \Delta x(T) \) and \( \Delta y(T) \). The iteration to successive varied trajectories is continued until the terminal errors are acceptable. A suitable initial guess for the angle \( \alpha \) is the inclination angle of the straight line from \( A \) to \( B \).

5. Numerical example

Ten successive semidaily analyses of wave height and direction, starting at 06Z on 4 May 1963, were furnished by the Fleet Numerical Weather Facility. The maximum speed of the chosen P2-S2-R2 ship type is 19.6 knots. This combination of data precluded a trip of great length. It was decided to select an area of continued extreme wave height for the example. Such an area was found centering at 30° North latitude and 162° East longitude. Figs. 4, 5 and 6 show two computed minimal-time ship tracks in the area, with contours of wave height in feet and wave direction arrows. The arc traversed by the ship during the 6 hours preceding and/or the 6 hours following the time of each Figure is shown as a dashed curve. The minimal-time track \( AB \) required 2.488 days with a 3.0% saving over the geodesic track. The minimal-time track \( CD \) required 2.656 days with a 1.3% saving over the geodesic track. The severity of the sea conditions in the area preclude any more spectacular saving. The highly non-analytic nature of the wave height in the area was found to affect the convergence of the Newton-Raphson iteration of (22). It was found necessary to halve the values of \( \Delta T \) and \( \delta \alpha \) in order to avoid a divergent oscillation. Resort to this stratagem was found to be unneces-
Fig. 4. Sea conditions
nd 182 on 4 May 1963.
Fig. 5. Sea conditions
nd 18Z on 5 May 1963.
id 182 on 6 May 1963.
sary when the wave height was more nearly analytic.

6. Concluding remarks

The numerical integrations involved in the theory of minimal-time ship routing through time-dependent wave fields are found to be feasible. The necessary three-dimensional interpolations in the wave field data, discussed in the Appendix, present no problem. Convergence problems may arise, but can be solved by the described delayed approach to the limit. The authors can supply copies of their Fortran programs for ship routing and for the cubic-interpolation contouring of Figs. 4, 5 and 6.

Acknowledgements. This work was supported by the Office of Naval Research. Aid from the U. S. Navy Fleet Numerical Weather Facility and from the Computer Facility, U. S. Naval Postgraduate School, is acknowledged.

7. Appendix

Some pertinent mathematical details are listed here.

The geographic wave direction \( \mathbf{v} \), measured clockwise from the North must be converted to the unit vector \( \mathbf{i}\cos\theta+j\sin\theta \) in the stereographic grid system by

\[
\begin{align*}
\cos K &= \frac{-(x-31)\cos\psi+(y-31)\sin\psi}{r} \\
\sin K &= \frac{-(x-31)\sin\psi-(y-31)\cos\psi}{r}
\end{align*}
\]

where \( r^2 = (x-31)^2 + (y-31)^2 \). Then the derivative

\[
K_x = \cos K \left( \frac{\sin K}{\partial x} \right) - \sin K \left( \frac{\partial \cos K}{\partial x} \right)
\]

The derivatives \( V_x = m\nu_x + v m_x \quad \text{and} \quad V_p = m\nu_p \) are obtained most conveniently by the implicit differentiation of the equation

\[
\left[ v \sin(p-K)/b \right]^2 + \left[ (c + v \cos(p-K))/a \right]^2 = 1
\]

of the elliptical polar velocity diagram, and noting that \( a, b, c \) are functions of \( H(x,y,t) \), and that \( K \) depends on \( x, y, t \). The com-
plexity of the result is reduced by introducing the parameter $\alpha$ defined by

$$\begin{align*}
\sin \alpha &= b \sin(q-K)/s = \n \sin(p-K)/b \\
\cos \alpha &= \cos(q-K)/s = \n \cos(p-K)+c)/a
\end{align*}$$

(26)

where $s^2 = a^2 \cos^2(q-K) + b^2 \sin^2(q-K)$.

The numerical integration of the adjoint Eqs. (7) and (8) demands an interpolation formula for $H(x,y,t)$, $\cos K(x,y,t)$ and $\sin K(x,y,t)$ which guarantees the continuity of these functions and of their first space and time derivatives where any of $x,y,t$ assume grid values. A 16-point interpolation formula to accomplish this is obtained from the $4 \times 4$ matrix $P$, whose four rows and columns of function entries correspond to four successive $x$ and $y$ grid values respectively. The interpolation mesh cell is the central cell of the array, with $x$ and $y$ measured from the cell center, and with the mesh distance considered to be two units. The formula is

$$F(x,y) = P(x)P' (y)/256$$

(27)

where the matrix

$$P(x) = [(1-x)(x^2-1),(x-1)(2x^2+2x-9),(x+1)(9+2x-3x^2),(x+1)(x^2-1)]$$

(28)

and the prime indicates matrix transposition. Interpolation in the time dimension is accomplished by the similar 2-unit-mesh central-difference formula

$$F(t) = [P(-3), P(-1), P(1), P(3)]P'(t)/16$$

(29)

which guarantees the continuity of $F(t)$ and $dF/dt$ at each end of the central time interpolation mesh. This formula is consistent with parabolic interpolation at the beginning or end of a time series, where central differences are not available. An interpolated vector $i \cos K(x,y,t) + j \sin K(x,y,t)$ should be normalized before use.
REFERENCES


FIGURE LEGENDS

Fig. 1. Polar velocity diagram.

Fig. 2. Ship speed in following, beam and head waves.

Fig. 3. Ship motion in stereographic Oxy plane.

Fig. 4. Sea conditions at 06Z and 18Z on 4 May 1963.

Fig. 5. Sea conditions at 06Z and 18Z on 5 May 1963.

Fig. 6. Sea conditions at 06Z and 18Z on 6 May 1963.
..JOB..B.E.I.C.K - STAFF BOX 6 - 3 MINUTE EXPRESS - 7/7/5/64

C YVARS(1)=XLM1 YVARS(2)=XLM2 YVARS(3)=XLM2 YVARS(4)=XLM2 JJ=0 EW
C YVARS(5)=X YVARS(6)=Y YVARS(7)=XJ YVARS(8)=S JJ=1 NS

DIMENSION KH(19,12,10),HT(19,12,10),KK(19,12,10),CSK(19,12,10),
+ SINK(19,12,10),DY(9),YC(8),CY(8),TAU(300),X(300),Y(300),
+ CAPLAN(300),PP(300),GQ(300),WX(300),VS(300),WH(300),XJ(300),
+ S(300),AK(4,8)

ECUVALENCE (KH,HT), (KK,CSK)

READ 1, KH, KK, JJ, LMAX, XXST, KYST, XXFN, KYFN, ALF, T, FAC, FMUL
1 FORMAT (55(38127), 60(38127), 613, 4F12.9)
PRINT 2, JJ, LMAX, XXST, KYST, XXFN, KYFN, FAC, FMUL
2 FORMAT (1H0, 613, 2F12.9)

DO 4 IX=1,9
DELI = 1.25
DO 4 IX=1,12
DELM = 0.1
ROOT = 5 CRTF(DELI*DELI + DELJ*DELJ)
DO 4 K=1,10
HT(I,J,K) = KH(I,J,K) * 10
ANGLE = ANGLE/57.2957 7951
COS = COSF(ANGLE)
SIN = SINF(ANGLE)

COSK(I,J,K) = (DELI*COS + DELJ*SIN)/ROOT
SINK(I,J,K) = (DELI*SIN - DELJ*COS)/ROOT

C(1) = 0.0
C(2) = 0.5
C(3) = 0.5
C(4) = 1.0
X(1) = XXST
Y(1) = KYST
TAU(1) = 0.0
XXFIN = XXFN
YYFIN = KYFN

WH(1) = HT(XXST, KYST, 1)
CALL POLAR (COSK(XXST, KYST, 1), SINK(XXST, KYST, 1), WK(1))

XSTEP = T/FAC
CALL ABC (WH(1), AI, BI, CI, DAI, DBI, DCI)

CAPLAN(1) = 1.0
S(1) = 0.0
XJ(1) = 0.0
DO 18 L=1, LMAX
Q = ALF
QQ(1) = C * 57.2957 7951
COSA = COSF(ALF)
SINA = SINF(ALF)
COSQ = COSA
SINQ = SINA

COSQMK = COSQ*COSK(XXST, KYST, 1) + SINQ*SINK(XXST, KYST, 1)
SINQMK = SINK*COSK(XXST, KYST, 1) - COSQ*SINK(XXST, KYST, 1)
ABS = AI*COSQMK
ORD = BI*SINQMK
HYP = SQRTF(ABS*ABS + ORD*ORD)
SINB = ABS/HYP
COSB = ADS/HYP
VMAJ = AI*COSB - CI
VMIN = BI*SINB

VS(1) = SQRTF(VMAJ*VMAJ + VMIN*VMIN)
CALL POLAR (VMAJ, VMIN, PMK)
PP(1) = PMK + WK(1)

XVAR = 1.0
DO 5 S=1,8
YVARS(1) = 0.0
YVARS(2) = 1.0
YVARS(3) = X(1)
YVARS(4) = Y(1)
N1 = T/XSTEP + 1.0
XN1 = N1
STEP = T/XN1
N2 = N1 + 1
DO 14 K=2, N2
DO 7 I=1,4
X = XVAR + C(I)*STEP
DO 6 J=1, 8
Y(1) = YVARS(J) + C(I)*AK(I-1, J)

CALL TERP (HT, YC(5), YC(6), XC, H, MX, HY)
HYP = SCRTF((ABS*ABS + ORD*ORD)
SINB = ORD/HYP
gusb = ABS/HYP
VMAJ = FA*COS - FC
VMIN = FA*SIN
VS(K) = SCRTF(VMAJ*VMAJ + VMIN*VMIN)
COSP = (CK*VMAJ - SK*VMIN)/VS(K)
SINP = (SK*VMAJ + CK*VMIN)/VS(K)
call pocal((cosp, sinp, pp(k))
call pocal((cosq, sinq, qq(k))
call pocal((ck, sk, wk(k))
XJ(K) = YVARS(1)/YVARS(1) + YVARS(4) - YVARS(2) + YVARS(3)
S(K) = YVARS(2)
IF (N2-K) 82, 15, 82
82 IF (JJ) 88, 83, 89
83 IF (XFIN-X1), 85, 21, 84
84 IF (X(K)-FFIN) 86, 11, 11
85 IF (Y(K)- XFIN) 11, 11, 86
86 IF (Y(K)-XFIN) 11, 11, 87
87 IF (1C0 Y(K)) 11, 11, 14
88 IF (YFIN-Y1) 90, 21, 89
89 IF (Y(K)-YFIN) 91, 11, 11
90 IF (Y(K)-YFIN) 11, 11, 91
91 IF (X(K)-3.00) 11, 11, 92
92 IF (16.0-X(K)) 11, 11, 14
11 N2 = K
12 T = TAU(k)
13 FORMAT (12HO EARLY N2 = 12/)
14 CONTINUE
15 PRINT 16
16 PRINT (1H06X2HNI212X3HALF14X1HT14X1X13X1HY)
PRINT 17, N2, ALF, T, X(N2), Y(N2)
17 FORMAT (110, 4F15.9)
DELX = X(N2) - 20.0
DELY = Y(N2) - 1.0
EMFI = (97.4, 75 + DELX*DELX + DELY*DELY)/1043.638 743
CAPV = VS(N2)*EMFI/18.702 181 818
XDOT = CAPV*COSP
YDOT = CAPV*SINP
FINJ = XIJ(N2)
DETER = FINJ*(XDOT*XLAM + YDOT*XMU)
DIFX = XFIN - X(N2)
DIFY = YFIN - Y(N2)
DIFT = FINJ*(XLAM*DIFX + XMU*DIFY)/DETER
DIFA = (XDOT*DIFT) - YDOT*DIFT)/DETER
T = T + FMUL*DIFT
ALF = ALF + FMUL*DIFA
PRINT 16
PRINT 17, N2, ALF, T, X(N2), Y(N2)
18 CONTINUE
19 PRINT 19
19 FORMAT (114X3HTAJ8X1HX9X1HY6X6HCAPLAM5X2HWH8X2HWK8X2HVS9X1HS8X2H)
+ X2HPPE2X2HQC/)
PRINT 20, (TATAU, X(K), Y(K), CAPLAM(I), WH(I), WK(I)),
+ VS(I), S(1), XIJ(I), PPI(I), QQ(I), 1 = 1, N2)
20 FORMAT (11F10.4)
21 STOP
END
SUBROUTINE ABC (H, A, B, C, DA, DB, DC)
RA1 = SCRTF((0.338 635 06466*H-0.492 435 9727)*H+2.099 345 872)
A = -0.120 197 666*H + 20.748 910 236 - RAD1
B = -0.126 879 519*H + 20.633 462 763 - RAD2
B = -0.126 879 519*H + 20.633 462 763 - RAD2
RAD4 = SCRTF((0.313 051 7910*H-0.193 709 6314*H+0.814 886 3262)
-END
-0.125 424 403*H - 0.293 444 8925 - RAD3 + RAD4
RA1 = -0.120 197 666*H - 0.038 635 06466*H - 0.246 217 98635)/RAD1
DB = -0.126 879 519 - 0.032 937 3068*H - 0.219 903 39595)/RAD2
DC = -0.126 424 493 + 0.010 441 89560*H - 0.034 239 69455)/RAD3
+ 0.013 051 79100*H - 0.096 845 81575/RAH4
END
SUBROUTINE POLAR (X, Y, P)
IF (X) 10, 3, 10
P = 0.0
RETURN
P = 0.0
RETURN
P = 90.0
RETURN
P = -90.0
RETURN
9 RETURN
10 P = 57.2957 7951 • ATANF(Y/X)
11 IF (X) 12, 14, 14
12 IF (Y) 15, 13, 13
13 P = P - 180.0
14 RETURN
15 P = P - 180.0
16 END
SUBROUTINE TERP (FUNC, X, Y, T, OUT, CUTX, CUTY)
DIMENSION FUNC(19, 12, 10), PT(4, 4), P(4), Q(4), PX(4), QY(4), ORD(4),
ORDX(4), ORDY(4), S(4), D(4)
LL = 1
IF (L) 1, 4
1 LL = 3
2 GO TO 4
3 LL = 2
4 M = XINTF(X) - 3
N = XINTF(Y) - 3
XX = 2.0 * (X - INTF(X)) - 1.0
YY = 2.0 * (Y - INTF(Y)) - 1.0
TT = 2.0 * (T - INTF(T)) - 1.0
XP1 = XX + 1.0
XM1 = XX - 1.0
YP1 = YY + 1.0
YM1 = YY - 1.0
TP1 = TT + 1.0
TM1 = TT - 1.0
X2M = XP1 * XM1
Y2M = YP1 * YM1
T2M = TP1 * TM1
P(1) = -XM1 * X2M
P(2) = ((3.0 * XX + 2.0) * XX - 9.0) * XM1
P(3) = -2.0 * XX * XX + 18.0 - P(2)
P(4) = XP1 * X2M
Q(1) = -YM1 * Y2M
Q(2) = ((3.0 * YY + 2.0) * YY - 9.0) * YM1
Q(3) = -2.0 * YY * YY + 18.0 - Q(2)
Q(4) = YP1 * Y2M
PX(4) = (3.0 * XX - 1.0) * XP1
PX(1) = 4.0 * XX - PX(4)
PX(2) = (9.0 * XX - 11.0) * XP1
PX(3) = -4.0 * XX - PX(2)
QY(4) = (3.0 * YY - 1.0) * YP1
QY(1) = 4.0 * YY - QY(4)
QY(2) = (9.0 * YY - 11.0) * YP1
QY(3) = -4.0 * YY - QY(2)
CC 6 K = 1, 4
DO 5 J = 1, 4
DO 5 K = 1, 4
5 PT((I, J) = FUNC(M+1, N+J, L+K)
DO 10 I = 1, 14
9 S(I) = P(I) * PT((I, 1)) + P(2) * PT((2, 1)) + P(3) * PT((3, 1)) + P(4) * PT((4, 1))
10 D(I) = C(I) * PT((I, 1)) + Q(2) * PT((2, 1)) + Q(3) * PT((3, 1)) + Q(4) * PT((4, 1))
ORDX(K) = (D(1) * PX(1) + D(2) * PX(2) + D(3) * PX(3) + D(4) * PX(4)) / 128.
ORDY(K) = (S(1) * QY(1) + S(2) * QY(2) + S(3) * QY(3) + S(4) * QY(4)) / 128.
IF (LL = 2) 8, 9
7 G = (3.0 + TT + 2.0) * TT - 9.0
OUT = (TM1 * G * ORD(2) - T2M * ORD(1)) + TP1 * (M * ORD(3) - T2M * ORD(4)) / 16.
OUTX = (TM1 * G * ORDX(2) - T2M * ORDX(1)) + TP1 * (M * ORDX(3) - T2M * ORDX(4)) / 16.
OUTY = (TM1 * G * ORDY(2) - T2M * ORDY(1)) + TP1 * (M * ORDY(3) - T2M * ORDY(4)) / 16.
RETURN
8 TM3 = TT - 3.0
OUT = (TM3 * (TM1 * ORDI(1) - 2.0 * TP1 * ORDI(2)) + T2M * ORDI(3)) / 8.0
OUTX = (TM3 * (TM1 * ORDI(1) - 2.0 * TP1 * ORDI(2)) + T2M * ORDX(3)) / 8.0
OUTY = (TM3 * (TM1 * ORDI(1) - 2.0 * TP1 * ORDI(2)) + T2M * ORDY(3)) / 8.0
RETURN
9 TP3 = TT + 3.0
OUT = (TP3 * (TP1 * ORDI(4) - 2.0 * TM1 * ORDI(3)) + T2M * ORDI(2)) / 8.0
OUTX = (TP3 * (TP1 * ORDI(4) - 2.0 * TM1 * ORDI(3)) + T2M * ORDX(2)) / 8.0
OUTY = (TP3 * (TP1 * ORDI(4) - 2.0 * TM1 * ORDI(3)) + T2M * ORDY(2)) / 8.0
END
TO CONVERT PROGRAM INVPCY TO PROGRAM STRAIGHT
(1) REPLACE MINVOY CARDS (43-55) BY FOLLOWING CARDS
   COST = -SINK(KKST, KYST, 1)
   OR COST = COS(KKST, KYST, 1)
   SINT = -COST(KKST, KYST, 1)
   OR SINT = -SINK(KKST, KYST, 1)
   QUAD = BI*R*CO*ST*COST + AI*AI*SINT*SINT
   RAD = SQRTR(QUAD - (C1*SINT)**2)
   SINB = SINT*(AI*ADI - BI*C1*COST)/QUAD
   COSB = (BI*COST*ADI + AI*C1*SINT*SINT)/QUAD
(2) REPLACE MINVOY CARDS (89-97) BY FOLLOWING CARDS
   COST = -SK
   OR COST = CK
   SINT = -SK
   OR SINT = -SK
   QUAD = FB*FC*COST*COST + FA*FA*SINT*SINT
   RAD = SQRTR(QUAD - (FC*SINT)**2)
   SINB = SINT*(FA*ADI - FB*FC*COST)/QUAD
   COSB = (FB*COST*ADI + FA*FC*SINT*SINT)/QUAD
(3) REPLACE MINVOY CARDS (153-161) BY FOLLOWING CARDS
   COST = -SK
   OR COST = CK
   SINT = -SK
   OR SINT = -SK
   QUAD = FB*FC*COST*COST + FA*FA*SINT*SINT
   RAD = SQRTR(QUAD - (FC*SINT)**2)
   SINB = SINT*(FA*ADI - FB*FC*COST)/QUAD
   COSB = (FB*COST*ADI + FA*FC*SINT*SINT)/QUAD
(4) REPLACE MINVOY CARDS (197-198) BY FOLLOWING CARDS
   FF = (5.00-Y(N2-1))/(Y(N2)-Y(N2-1))
   OR FF = (13.0-X(N2-1))/(X(N2)-X(N2-1))

   T = FF*TAU(N2) + (1.0-FF)*TAU(N2-1)

PROGRAM CONTOUR
DIMENSION ABS(900), ORD(900), AH(19,12), HT(19,12), KK(19,12), PT(12)
+COSK(19,12), SINK(19,12), RX(48), RY(48), CON(4,4), IT(12), TI(12), O(3)
+EQUIVALENCE (IT, TI), (LA, AL)
READ 1, DS, IHMAX, IDELH,IXMIN, IXMAX, IYMIN, IYMAX, KH, KK, TI, AL
1 FORMAT (F4,2, 613, 12(38), 2(10A)/)
DO 2 J=1, 12
DO 2 I=1, 19
2 PRINT 3, DS, IHMAX, IDELH, IXMIN, IXMAX, IYMIN, IYMAX
3 FORMAT (1HO, F4, 2, 613 )
READ 26, (ABS(I), ORD(I), I=1,5)
26 FORMAT (10F3.1)
CALL CRAY (5, ABS, ORD, 1, 0, LA, IT, 1, 1, 0, 0, 2, 2, 8, 8, 0, LAST)
C = DS*DS
DO 22 IY = IYMIN, IYMAX
   Y = IY
   DO 22 IX = IXMIN, IXMAX
      X = IX
      CALL COEF (HT, IX, IY, PT, CON)
   DO 22 I = 3, IHMAX, IDELH
      H = IH
      CALL ROOT (CON, 1, H, D, KER)
      IF (KER) 8, 8, 4
   4 DO 7 J=1, KER
      ABS(1) = X + (D(J)+1.0)/2.0
      ORD(1) = Y
      IF (NR) 33, 35, 33
   33 DO 34 I=1, NR
         A = ABS(1) - RX(1)
         R = ORD(1) - RY(1)
         IF (A*A + B*B - C) 7, 7, 34
   34 CONTINUE
   35 CALL GRAD (PT, ABS(1), ORD(1), DX, DY, QUAD, DELX, DELY, DS, H, 0)
   35 IF (QUAD - 1.0 < 10) 7, 7, 5
   5 IF (DELY) 6, 36, 36
   6 DS = -DS
   6 DELX = DX - DX
   6 DELY = DELY - DY
   6 IF (DELX) 7, 36, 36
   36 ABS(2) = ABS(1) + DELX
   36 ORD(2) = ORD(1) + DELY
   CALL CUT (NH, ABS, ORD, RX, RY, DS, PT, X, Y, H)
   7 CONTINUE
   7 IF (KER) 13, 13, 9
   8 CALL ROOT (CON, 2, H, D, KER)
   9 DO 12 J=1, KER
12 CONTINUE
- 14 -
SIN = SIN((IX, IY))
ABS(1) = X + (CS/3.0)
ORD(1) = Y * SIN/3.0
ABS(2) = X * X - ABS(1)
ORD(2) = Y - ORD(1)
ABS(3) = ABS(2) + (1.732050807 * COS - SIN)/18.0
ORD(3) = ABS(3) + (1.732050807 * SIN + COS)/18.0
ABS(4) = ABS(3) + SIN/9.0
ORD(4) = ORD(3) - COS/9.0
ABS(5) = ORD(4)
ORD(5) = ORD(2)
DO 28 I = 1, 5
ABS(I) = 0.8 * ABS(I) - 4.0
ORD(I) = 0.8 * ORD(I) - 3.2
CALL ERAW (52, ABS, ORD, 3, 0, LA, IT, 1, 1, 0, 2, 8, 0, LAST)
END
SUBROUTINE COEF (HT, IX, IY, P, CON)
DIMENSION HT(19, 12), P(12), CON(4, 4)
P(1) = HT(IX, IY - 1)
P(2) = HT(IX+1, IY - 1)
LX = IX - 2
DO 1 I = 1, 4
P(2+I) = HT(LX+I, IY)
P(6+I) = HT(LX+I, IY+1)
P(11) = HT(IX, IY+2)
P(12) = HT(IX+1, IY+2)
CON(1, 1) = P(6) - P(3) + 3.0 * (P(4) - P(5))
CON(2, 1) = P(3) - P(4) - P(5) + P(6)
CON(3, 1) = P(3) - P(6) + 11.0 * (P(5) - P(4))
CON(4, 1) = -P(3) - P(6) + 9.0 * (P(5) + P(4))
CON(1, 2) = P(10) - P(7) + 3.0 * (P(8) - P(9))
CON(2, 2) = P(10) - P(9) - P(8) + P(7)
CON(3, 2) = P(7) - P(10) + 11.0 * (P(9) - P(8))
CON(4, 2) = -P(7) - P(10) + 9.0 * (P(9) + P(8))
CON(1, 3) = P(11) - P(1) + 3.0 * (P(4) - P(8))
CON(2, 3) = P(11) - P(4) - P(8) + P(11)
CON(3, 3) = P(11) - P(11) + 11.0 * (P(8) - P(4))
CON(4, 3) = -P(11) - P(11) + 9.0 * (P(8) + P(4))
CON(1, 4) = P(12) - P(2) + 3.0 * (P(5) - P(9))
CON(2, 4) = P(12) - P(9) - P(5) + P(2)
CON(3, 4) = P(2) - P(12) + 11.0 * (P(5) - P(9))
CON(4, 4) = -P(2) - (12) + 9.0 * (P(5) + P(9))
END
SUBROUTINE ROOT (CON, KS, H, D, KER)
DIMENSION CON(4, 4), D(3), X(3)
FOURTH = CON(4, KS) - 16.0
IF (CCN(1, KS)) 12, 2, 12
IF (CCN(2, KS)) 6, 3, 6
IF (CCN(3, KS)) 11, 4, 11
IF (FOURTH) 9, 5, 9
D(1) = -1.0
D(2) = 1.0
KER = 2
RETURN
B = 0.5 * CON(3, KS) / CON(2, KS)
C = FOURTH / CON(2, KS)
QUAD = B * B - C
RAD = SQRTF(ABS(SQRT(QUAD)))
IF (QUAD) 7, 8, 10
IF (RAD) 1.6E-5, 8, 8, 9
X(1) = -B
KER = 1
GO TO 28
KER = 0
RETURN
X(1) = -B + RAD
X(2) = -X - RAD
KER = 2
GO TO 28
11 X(1) = -FOURTH/C3N(3, KS)
KER = 1
GO TO 28
12 A = C3N(2, KS)/C3N(1, KS)
B = C3N(1, KS)/C3N(1, KS)
P = A*A/9.0 - B/3.0
Q = A*B/6.0 - C/2.0 - A*A*A/27.0
RAD = SQRT(ABSF(A))
IF (P) GOTO 13, 16, 18
13 PHI3 = ASINH(Q/(RAD*RAD*RAD))/3.0
X(1) = 2.0*RAD*SINH(PHI3) - A/3.0
KER = 1
IF (1.732050807*RAD*COSH(PHI3) - 1.E-5) 15, 15, 28
15 X(2) = -(X(1)+A)/2.0
KER = 2
GO TO 28
16 X(1) = [2.0*ABSF(Q)]**3.33333333333333 - A/3.0
KER = 1
IF (Q) GOTO 17, 28, 28
17 X(1) = -X(1) - A/1.5
GO TO 28
18 ARG = Q/(RAD*RAD*RAD)
IF (ABSF(ARG) < 1.0) 27, 25, 19
19 PHI3 = ACOSF(ABSF(ARG))/3.0
X(1) = 2.0*RAD*COSF(PHI3) - A/3.0
KER = 1
IF (1.732050807*RAD*SINF(PHI3) - 1.E-5) 15, 15, 28
21 X(1) = 2.0*RAD*COSF(PHI3) - A/3.0
KER = 2
GO TO 22
22 X(1) = 2.0*RAD*COSF(PHI3) - A/3.0
KER = 3
GO TO 28
25 X(1) = 2.0*ABSF(Q)**3.33333333333333 - A/3.0
X(2) = -(X(1)+A)/2.0
KER = 2
IF (Q) GOTO 26, 28, 28
26 X(2) = -X(2) - A/1.5
GO TO 17
27 PHI3 = ACOSF(ARG)/3.0
SIN = 1.732050807*RAD*SINF(PHI3)
Z = RAD*COSF(PHI3)
X(1) = Z + Z - A/3.0
X(3) = Z + SIN - A/3.0
X(5) = X(2) - SIN - 1
KER = 3
GO TO 28
28 JAR = 0
DO 30 I=1,KER
IF (ABSF(X(I)) < 1.0) 29, 29, 30
29 JAR = JAR + 1
D(JAR) = X(I)
30 CONTINUE
KER = JAR
END
SURROUTINE GRAD (P, X, Y, DX, DY, QUAD, DELX, DELY, DS, H, KC)
DIMENSION P(12)
XX = 2.0*(X - INTF(X)) - 1.0
YY = 2.0*(Y - INTF(Y)) - 1.0
XP1 = XX + 1.0
XM1 = XX - 1.0
YP1 = YY + 1.0
YM1 = YY - 1.0
TX = 2.0*XX
TY = 2.0*YY
CIR = 3.0*(XX*XX+YY*YY) - 10.0
X2M = XP1*XM1
Y2M = YP1*YM1
SXH = 6.0*XX + 2.0
SYM = 6.0*YY + 2.0
QXP = 18.0*XX + 2.0
QYM = 18.0*YY + 2.0
P1 = (TX*P(10) - SXH*P(9))*XP1 - 17 -