NOAA TR ERL 303-AOML 15

NOAA Technical Report ERL 303-AOML 15

U.S. DEPARTMENT OF COMMERCE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
Environmental Research Laboratories

A Connected Least-Squares Adjustment of Navigation Data

L.M. DORMAN
J.W. LAVELLE

BOULDER, COLO.
JULY 1974

ENVIRONMENTAL RESEARCH LABORATORIES

sion of the Environmental Research Laboratories is to study the oceans, inland waters, the lower and upper atmosphere, the space environment, and the earth, in search of the understanding needed to provide more useful services in improving man's prospects for survival as influenced by the physical environment. Laboratories contributing to these studies are:

Atlantic Oceanographic and Meteorological Laboratories (AOML): Geology and geophysics of ocean basins and borders, oceanic processes, sea-air interactions and remote sensing of ocean processes and characteristics (Miami, Florida).

Pacific Marine Environmental Laboratory (PMEL): Environmental processes with emphasis on monitoring and predicting the effects of man's activities on estuarine, coastal, and nearshore marine processes (Seattle, Washington).

Great Lakes Environmental Research Laboratory (GLERL): Physical, chemical, and biological limnology, lake-air interactions, lake hydrology, lake level forecasting, and lake ice studies (Ann Arbor, Michigan).

Atmospheric Physics and Chemistry Laboratory (APCL): Processes of cloud and precipitation physics; chemical composition and nucleating substances in the lower atmosphere; and laboratory and field experiments toward developing feasible methods of weather modification.

Air Resources Laboratories (ARL): Diffusion, transport, and dissipation of atmospheric contaminants; development of methods for prediction and control of atmospheric pollution; geophysical monitoring for climatic change (Silver Spring, Maryland).

Geophysical Fluid Dynamics Laboratory (GFDL): Dynamics and physics of geophysical fluid systems; development of a theoretical basis, through mathematical modeling and computer simulation, for the behavior and properties of the atmosphere and the oceans (Princeton, New Jersey).

National Severe Storms Laboratory (NSSL): Torandoes, squall lines, thunderstorms, and other severe local convective phenomena directed toward improved methods of prediction and detection (Norman, Oklahoma).

Space Environment Laboratory (SEL): Solar-terrestrial physics, service and technique development in the areas of environmental monitoring and forecasting.

Aeronomy Laboratory (AL): Theoretical, laboratory, rocket, and satellite studies of the physical and chemical processes controlling the ionosphere and exosphere of the earth and other planets, and of the dynamics of their interactions with high-altitude meteorology.

Wave Propagation Laboratory (WPL): Development of new methods for remote sensing of the geophysical environment with special emphasis on optical, microwave and acoustic sensing systems.

Marine EcoSystem Analysis Program Office (MPO): Plans and directs interdisciplinary analyses of the physical, chemical, geological, and biological characteristics of selected coastal regions to assess the potential effects of ocean dumping, municipal and industrial waste discharges, oil pollution, or other activity which may have environmental impact.

Weather Modification Program Office (WMPO): Plans and directs ERL weather modification research activities in precipitation enhancement and severe storms mitigation and operates ERL's research aircraft.

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
BOULDER, COLORADO 80302

U.S. DEPARTMENT OF COMMERCE
Frederick B. Dent, Secretary

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
Robert M. White, Administrator
ENVIRONMENTAL RESEARCH LABORATORIES
Wilmot N. Hess, Director

NOAA TECHNICAL REPORT ERL 303-AOML 15

A Connected Least-Squares Adjustment of Navigation Data

L.M. DORMAN
J.W. LAVELLE

BOULDER, COLO.
July 1974

For sale by the Superintendent of Documents, U. S. Government Printing Office, Washington, D. C. 20402

DISCLAIMER

The Environmental Research Laboratories do not approve, recommend, or endorse any proprietary product or proprietary material mentioned in this publication. No reference shall be made to the Environmental Research Laboratories, or to this publication furnished by the Environmental Research Laboratories, in any advertising or sales promotion which would indicate or imply that the Environmental Research Laboratories approve, recommend, or endorse any proprietary product or proprietary material mentioned herein, or which has as its purpose an intent to cause directly or indirectly the advertised product to be used or purchased because of this Environmental Research Laboratories publication.

CONTENTS

A CONNECTED LEAST-SQUARES ADJUSTMENT OF NAVIGATION DATA

L. M. Dorman* and J. W. Lavelle

ABSTRACT

An algorithm is described that will determine the position of a ship in the least-squares optimal sense, given infrequent and irregularly spaced estimates of the ship's position and information on the ship's attempted course and speed. The requirements imposed on the solution are: (1) that it be continuous; (2) that it be continuous in the first derivative except at points of ordered intentional speed and course change (e.g., eliminating fictitious discontinuities in the Eötvös correction for gravity data); and (3) that it accommodate short-period small adjustments to an overall course. These constraints, coupled with sufficient procedural flexibility to allow user intervention in the ultimate determination of the track's parametrization, make this scheme a useful tool in geophysical navigation work. To emphasize its practical nature and ease of application, we have provided an example of the algorithm's use as well as an appendix which describes the content and use of subroutines written to automate the procedure.

INTRODUCTION

One of the most vexing problems encountered in the preparation of geophysical data taken at sea is the determination, with adequate accuracy, of the ship's position when a measurement is made. Because of the necessity of relying on position data sampled at irregular intervals and with irregular consistency, a reliable and straightforward method is needed to weigh each of the position measurements and to reduce them to a "smooth" track over the ocean surface. It is the

---

*Now at Scripps Institution of Oceanography, La Jolla, Calif. 92037

construction of an estimate of a ship's track, which is in some sense optimal, that this report addresses.

The most accurate navigation control generally available to the scientific community in the open ocean is that from the U.S. Navy satellite navigation system (Guier, 1966) which provides fixes at irregular intervals whose average is 100 min or so. The position so determined is said to have a one-quarter-mile standard deviation. Talwani (1970) reviewed the systems available for interpolation between satellite fixes and discussed adjustment methods. The simplest and most widely used technique is to add a constant correction-velocity vector to the dead-reckoned (DR) track so that the DR position at the time of the second fix coincides with the location of that fix. This allows one to include variations of the ship's course and speed that have shorter periods than several hundreds of minutes. The drawback of this procedure is that the correction velocities will be different for each fix pair so that the first derivative of the track, which in gravity data controls the Eötvös correction, will be discontinuous at the fixes -- a situation which may at times be gravely unrealistic.

Bowin et al. (1972) linearly interpolated the velocity between the center points of fix pairs to eliminate these discontinuities. Hayes, Talwani, and Worzel (personal communication, 1973) fitted polynomials to the cumulative north and east sets for single rhumb lines, obtaining a correction vector which varies smoothly with time in an effort to eliminate the unreal first-order discontinuity problem.

In an entirely different approach toward improved navigation, Johnson (1971) demonstrated how observed bathymetric or gravity fields in the vicinity of track crossings may be used to improve the estimates of the ship's position. Although this self-consistency approach can be valuable, the difficulties associated with automating the procedure and the dependence on the observed field will likely limit its application as a general-purpose tool. The algorithm we present may be of some help with that problem, however, as the parametrization of the corrections used by Johnson is similar to our own. In fact, depending on the need for additional constraints and the tractability of the observed fields, it may be possible to develop a hybrid procedure based on the inclusion of his crossing equations in our observational system.

What we present here is an improved navigation method without the problem of fictitious velocity jumps. We adjust a track so that the long wavelength features of the track are controlled by the satellite (or other) navigation system while the short wavelength features are taken from the DR input. The adjustment process preserves the continuity of the track (a connected track) and allows discontinuities in the velocity (and hence in the Eötvös correction) only at points of <u>actual</u> speed or course change. Finally, the adjustment process is flexible enough to allow user intervention in the decision as to when additional correction parameters are to be used. This later consideration allows the track to follow the navigation input more closely (accept shorter wavelength corrections) when fixes are frequent and reliable, but accepts only longer wavelengths when fixes are infrequent or unreliable.

SOLUTION

In the following treatment, we will work with a DR track which is constructed using ordered or estimated courses and speeds because these are most commonly available, although a DR from log and compass can be as easily used. We are also considering the case of a ship steering a constant course (a rhumb line) over some interval of time and then turning to steer another rhumb line. This is the most common occurrence in geophysical survey work and provides us with the simplicity guaranteed by the conformal Mercator projection, dealing with the tracks in a rectangular Cartesian coordinate system. The only disadvantages are those of a latitude-dependent scale.

The core of our suggested procedure is to make use of an explicit dependence of the ship's position on time, a technique unavailable to the manual smooth plotter. In Cartesian coordinates, the east (X) and north (Y) components of the two-dimensional vector function of time describing the ship's position can be treated separately. Our problem then becomes one of fitting two functions $\hat{X}(t)$ and $\hat{Y}(t)$ to the x- and y-components of the available fixes. To satisfy our objectives concerning the wavelength of the adjustments that we are willing to make, we let $\hat{X}(t)$ be the sum of the x-component of the DR track $X_{DR}(t)$ and let a piecewise smooth correction $X(t)$, whose parameters we determine by using the method of least squares, make the corrected track $\hat{X}(t)$ pass as closely as possible to the fixes. The term "piecewise smooth" means having a continuous derivative except at a few points. The y-component is treated identically, so we will discuss only the x-component.

Before we proceed further, let us define some terms and notation which will become useful.

A DR line is a track segment over which the ordered course and speed remain constant.

A correction line is a track segment over which the winds and currents are expected to be constant or to vary slowly. It may consist of more than one DR line if only small course changes have been made, or it may be only part of a DR line if there is a sufficient number of fixes on the whole line. In short, it is a line on which a single correction velocity and acceleration are operative.

A parameter is a polynomial coefficient representing a position, velocity, or acceleration correction which is active for part or all of a correction line.

TB(J) is the start time of the Jth parameter.

TE(J) is the end of the active time of the Jth parameter.

IT(J) is the type of the Jth parameter. The Type 1 parameter (of which there is only one per data set) is the x-component of the origin of the trackline. The Type 2 parameter is a constant velocity correction acting over the time interval delimited by TB and TE. The Type 3 parameter is a constant acceleration acting over the time interval delimited by TB and TE. This notation is chosen to preserve symmetry in the solution.

When computing a correction at time TT, the effective time for a parameter is zero if $TT \leq TB$; it is TT-TB if $TB < TT \leq TE$; and it is TE-TB if $TT > TE$. In the terminology of set theory, it is the intersection of the sets [0, TT] and [TB, TE].

To insure the continuity of the correction vector, we let X(t) be obtained by integrating the x-components of the various correction velocities and accelerations acting since the time origin of the problem. At times of speed changes or major course changes, where the effect of the wind and seas on the ship's motion can be markedly altered, we require only continuity of position. At those times, we allow the correction velocity to change. Along a track segment where we expect the set to vary slowly, we allow flexibility in the correction by letting a new correction acceleration to be added from time to time, thus allowing curves to be added to the trackline while maintaining continuity of velocity. In the case where the fixes along a track segment are insufficient to determine a correction velocity or acceleration, we can accept the DR track by simply not adding any correction velocities or accelerations.

The function X(t) is thus similar to the spline function in that it is piecewise polynomial; but it differs because it will generally be a least-squares approximation instead of an interpolation and because the continuity constraint is not the same at all junction points of the polynomial segments.

We demonstrate how the function X(t) is constructed by using as an example a portion of a real survey. The fixes and the adjusted trackline are shown in figure 1, and the navigation input is shown in figure 2. A detailed explanation of the data input is given in the appendix.

The parameters of the correction function X(t), which we must determine by our least-squares fit, are contained in the vector $\underline{B}$ that we determine by solving the matrix equation

$$\underline{A} \times \underline{B} = \underline{C} + \underline{E}. \tag{1}$$

This matrix equation is composed of one scalar equation for each fix in the data set. $\underline{C}$ is a vector whose i^{th}-component is the x-coordinate of the i^{th}-fix minus the x-coordinate of the DR position at the time of the fix. $\underline{E}$ is a vector of observational error and $\underline{A}$ is the matrix of coefficients which we will develop shortly.

After reading in all the information comprising a set, the beginning, end, and number and types of parameter on each correction line are determined according to the following basic rules:

(1) The first parameter (and the only Type 1 parameter) will always be a position (the origin of the adjusted trackline).

(2) The current correction line is terminated and a new one is begun when there is a change in ordered speed or a major ($\geq 10°$) course change.

(3) Each correction line will have two new parameters, a velocity and an acceleration, except in the following circumstances:

> (a) If the number of fixes on a correction line is smaller than NDFA, a preset constant, the acceleration is omitted ($2 \leq$ NDFA).
>
> (b) If the number of fixes on a correction line is smaller than NDFV, a preset constant, input course and speed are accepted ($1 \leq$ NDFV $<$ NDFA).
>
> (c) New velocity or acceleration parameters can be added at any time the user desires to allow for more complicated curves in the correction. The new velocity and acceleration parameters act from their start times until the termination of the correction line to which they belong.

Having determined the number, types, and start and stop times for all parameters in $\underline{B}$, the next step is to generate the matrix $\underline{A}$. Each row of $\underline{A}$, when multiplied by $\underline{B}$, must give the integrated correction distance up to the time of the fix. The integrated correction distance for the Type 1 (position) parameter is simply the parameter itself. For the Type 2 (velocity), it is the parameter multiplied by the effective time; and for the Type 3 (acceleration), it is one-half the parameter times the effective time squared.

In the example (see figs. 1 and 2), the first equation representing the first fix is

$$B(1) + B(2)\ (TT-TB[2]) + B(3)\ (TT-TB[3])^2/2 = X_{Fix1} - X_{DR}\ (TT) \qquad (2)$$

where TT is the time of the fix. The times TB [2] and TB [3] are equal to the start time of the line. In our example, the first line has seven fixes so that the first seven equations are of this form, differing only in the value of TT and X_{Fix}. X_{Fixi} is the x-component of the DR position at the time of the i^{th}-fix.

The eighth fix is on the second correction line, and its equation is

$$B(1) + BB(2)\ (TE[2]-TB[2]) + B(3)\ (TE[3]-TB[3])^2/2 + B(4)\ (TT-TB[4]) + B(5)\ (TT-TB[5])^2/2 = X_{Fix8} - X_{DR}\ (TT) \qquad (3)$$

where TE[2], TE[3], TB[4], and TB[5] are the times of the first course change. The redundancy in variable names preserves a symmetry which is useful in setting up the algorithm numerically. The generalization of these equations for the i^{th}-fix and M^{th}-correction line is straightforward, although we avoid writing down an explicit expression because of the notational complexity involved. In any case, we continue as in the above equations until we have generated all the rows of $\underline{A}$. We restate that the coefficient of the $\underline{B}$'s represents elements in the $\underline{A}$ matrix.

One will note that $\underline{A}$ is a "number of parameters" by the "number of fixes" dimensional matrix. This is the observational matrix described by Lanczos (1956). To obtain the square matrix which represents the normal equations, we premultiply by $\underline{A}$ transpose

$$[\underline{A}^T \times \underline{A}] \times \underline{B} = \underline{A}^T \times \underline{C} + \underline{A}^T \times \underline{E}. \qquad (4)$$

Because the error moments vanish in a least-square solution,

$$\underline{A}^T \times \underline{E} = 0, \qquad (5)$$

one is left with the equation

$$[\underline{A}^T \times \underline{A}] \times \underline{B} = \underline{A}^T \times \underline{C} \qquad (6)$$

which is a nonsingular square matrix premultiplying the column matrix of unknowns $\underline{B}$, that is, the standard least-squares form.

We would like to make one remark about the structure of $[\underline{A}^T \times \underline{A}]$. Because a trackline must "forget" a fix that is far removed from it in time, the matrix $[\underline{A}^T \times \underline{A}]$ must be diagonally dominant, that is, have elements which diminish in size away from the diagonal. This is the matrix analog of a fading memory. This is useful information to keep in mind when one deals with many days of navigational fixes. If such information is broken down into overlapping sets, one will be able to find overlapping trackline solutions with little difficulty.

After the values of the correction parameters have been obtained in this manner, one may go back to equations of the form of Eqs. (2) and (3) to generate X(t), the least-squares optimal track at any time t. What we usually do is to generate a track solution at frequent intervals (say 5 min) and write these down as the end product of the calculation. One will, of course, also find the positions of turning points, residuals of fixes from the track, and standard deviations of the corrector estimates. At the same time, we calculate the first derivative of the track, and an Eötvös correction is generated from the east-west velocity components. The adjusted track and the Eötvös corrections can then be combined with the observed data in the usual manner and can be plotted as in figure 1.

REAL DATA EXAMPLE

The example plotted in figure 1 treats a number of navigational problems which arose in a survey conducted in 1972 by the NOAA ship Discoverer near the Puerto Rico Trench. Measurements were begun at 0050 hr on Julian Day 100 (100/0050), which is the southernmost east-west line in the plot. We have set the algorithm constants so that two fixes are required before a velocity correction is computed and four fixes are required before an acceleration term is found.

Because the first line encompasses seven fixes, both velocity and acceleration corrections are calculated as well as the initial position of the ship. This last number is, of course, unique to the first line of any trackline sequence, as the initial position of subsequent lines is constrained to the end position of the previous line. Please note that there is one inflection point on this line at 100/0130 when the ship made a 3° course change. Because we are looking for the long wavelength corrections, we solve for only one set of correction parameters.

At 0830, a turn to the north was made and a second line begun; at that time, only a velocity correction was required as only three fixes were taken. At 1320, the ship turned to the west to find the longitude of the next survey line and then turned north at 1415 to find the proper latitude. Because of the relatively short time to complete these manuevers, fixes are sparse on one of these segments. While this is a vexation in manual smooth plotting, our procedure tolerates this situation.

We point out the following about lines of sparse data. There will be no velocity correction to the east-west line at 1320, so the length and orientation of this line are fixed by the lapsed time of the track and by the DR information. Therefore, a good approximation to the ship's course and speed should be input for this segment. This is not hard to do. We have found that an efficient way to handle this problem is to run the program with log speed entered as the first speed approximation. On lines that have more than the minimum number (NDFV) of fixes for determination of a velocity correction, the output will suggest how the log speed is likely modified in the face of prevailing current. This suggested correction to log speeds provides information for a better approximation of actual speed for weakly determined lines. The iterative second run through the procedure with the updated speeds provides a good estimate to the actual track. At 1615, the ship turned west to begin reflection profiling, but, as a result of hardware problems resulting in a poor record, was forced into retracing its course back to and beyond the start point. While the retracing allowed time to solve the hardware problems, the repositioning resulted in two short tracklines, one of which was poorly determined by satellite information. That line again is fixed in length and direction, but may translate in a way such that the overall sum of squares of residuals is minimum.

At 1915, with survey hardware operational, the ship proceeded for a period of 18 hr. Reading figure 2, one will note that small course corrections were made frequently (denoted by arrows in fig. 1), but it is not until 101/1300 that a course change of sufficient magnitude was made to terminate the ongoing correctors. In this case, without intervention, the entire line from 100/1915 to 101/1300 would have been fitted with a single velocity and a single acceleration parameter. This was judged to be insufficient in view of the apparent nonconstant and appreciable current along the track. For this reason, new acceleration parameters were introduced at 101/0240 and at 101/0700 to increase the flexibility of the fit. These interventions show as NA cards in figure 2. One will note that the number of fixes between the NA's and between the NA's and the end of the line is at least equal to NDFA (in this case, four).

As an aside, one should also note that this line is parallel to the east-west line below it, although they were run 24 hr apart. In addition, the two longest north-south lines exhibit a parallelism, all of which suggests the occurrence of a localized current running to the southwest.

At 1550, this east-west line was complete, and the ship slowed to bring in some of the streamed gear as the ship turned to the northeast to begin a quick run to the next survey line. Because of the large accelerations and decelerations at both ends of this line, we have decoupled this section of track from both the previous and following track solutions. This is evident as a gap in the line in figure 1.

At 101/1910, we began another line. We spare the reader detailed description of the line, which may be easily assembled from figures 1 and 2. Many features of the above description reappear on this track, suggesting that such navigational data are typical of a real survey and can be handled by the procedure.

As we have mentioned before, the memory of previous fixes retained by any line fades as the point of interest in time moves away from those fixes. This means that there is no real limitation to the number of tracks that the procedure will handle. If one has more fixes than can be handled by the available computer, then one may break the data into overlapping sets with the assurance that if the overlap were great enough, one would be able to find solutions that blend together in some region.

SUMMARY

We have suggested and demonstrated the usefulness of an algorithm which absorbs position measurements and ordered courses and speeds and calculates a connected smooth plot of the track. This procedure is based on quadradic-connected least-squares approximations for both latitude and longitude fixes explicitly parametrized by time. The resulting calculated track is continuous and assures one of continuous first derivatives (and hence Eötvös corrections), except at points of real speed or course change. Computer subroutines based on this algorithm and a description of their use are presented in the appendix. Because of the incompatibility of precise geophysical measurements and poor navigation, we think this procedure may contribute significantly to geophysical measurements taken at sea.

ACKNOWLEDGMENTS

We thank Stuart M. Smith for critically reading the manuscript. This work was funded in part by the National Science Foundation-IDOE Grant No. AG-253.

REFERENCES

Bowin, C., T.C. Aldrich, and R.A. Folinsbee (1972): VSA gravity meter system: tests and recent developments, *J. Geophys. Res. 77*:2018-2033.

Guier, W.H. (1966): Satellite navigation using integral Doppler data--the AN/SRN-9 equipment, *J. Geophys. Res. 71:*5903-5910.

Johnson, R.H. (1971): Reduction of discrepancies at crossing points in geophysical surveys, *J. Geophys. Res. 76*:4892-4896.

Lanczos, C. (1956): *Applied Analysis,* Prentice Hall Inc., Englewood Cliffs, N.J., 539 pp.

Talwani, M. (1970): Developments in navigation and measurement of gravity at sea, *Geoexploration 8*:151-183.

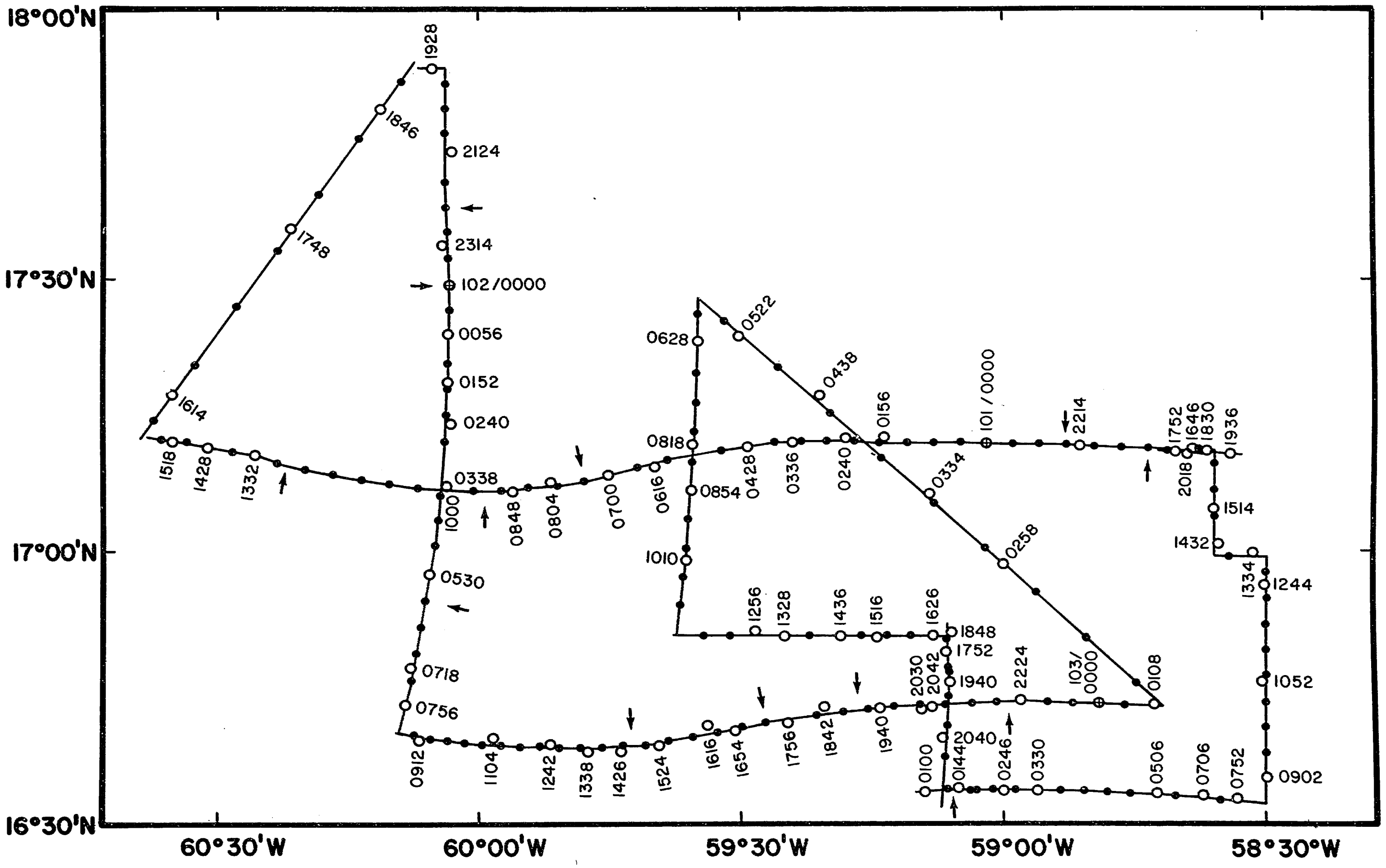


Figure 1. A plot of satellite fixes (open circles), with time annotation and a trackline based on them using the algorithm. Input information is listed in figure 2. Solid circles on the track have sectioned the track into 30-min intervals, and crossed circles have marked Julian Day midnights. The small arrows indicate points at which small course corrections were made.

DAY	TIME	LATITUDE	LONGITUDE	COURSE	SPEED	TYPE
100	0050			094	5	TP
100	0100	16 336	-59 091	094	5	SA
100	0130			097	5	TP
100	0144	16 339	-59 052	097	5	SA
100	0246	16 338	-58 598	097	5	SA
100	0330	16 337	-58 560	097	5	SA
100	0606	16 333	-58 423	097	5	SA
100	0706	16 330	-58 371	097	5	SA
100	0752	16 327	-58 332	097	5	SA
100	0830			002	5	TP
100	0902	16 350	-58 297	002	5	SA
100	1052	16 456	-58 299	002	5	SA
100	1244	16 562	-58 296	002	5	SA
100	1320			270	61	TP
100	1334	16 597	-58 309	270	61	SA
100	1415			004	55	TP
100	1432	17 009	-58 351	004	55	SA
100	1514	17 049	-58 353	004	55	SA
100	1615			269	55	TP
100	1640	17 113	-58 380	269	55	SA
100	1715			090	5	TP
100	1752	17 111	-58 397	090	5	SA
100	1830	17 113	-58 362	090	5	SA
100	1915			270	5	TP
100	1936	17 109	-58 337	270	5	SA
100	2018	17 109	-58 385	270	5	SA
100	2100			268	5	TP
100	2214	17 119	-58 509	268	5	SA
100	2230			266	5	TP
101	0156	17 125	-59 134	266	5	SA
101	0240			266	5	NA
101	0240	17 124	-59 178	266	5	SA
101	0336	17 122	-59 238	266	5	SA
101	0428	17 118	-59 292	266	5	SA
101	0616	17 096	-59 397	266	5	SA
101	0700			266	5	NA
101	0700	17 085	-59 451	266	5	SA
101	0730			273	5	TP
101	0804	17 075	-59 515	273	5	SA
101	0848	17 067	-59 560	273	5	SA
101	0915			277	5	TP
101	1000	17 071	-60 039	277	5	SA
101	1251			282	5	TP
101	1300			292	5	TP
101	1320			270	5	TP
101	1332	17 108	-60 257	270	5	SA
101	1428	17 115	-60 311	270	5	SA
101	1518	17 124	-60 351	270	5	SA
101	1550					
99999999						
101	1550			038	16	TP
101	1614	17 174	-60 351	038	16	SA
101	1740	17 357	-60 215	038	16	SA
101	1846	17 489	-60 113	038	16	SA
101	1910					
99999999						
101	1910			090	6	TP
101	1928	17 534	-60 054	090	6	SA
101	1940			180	5	TP
101	2124	17 441	-60 034	180	5	SA
101	2230			176	5	TP
101	2314	17 339	-60 040	176	5	SA
102	0000			180	5	TP
102	0056	17 241	-60 038	180	5	SA
102	0152	17 188	-60 037	180	5	SA
102	0240			180	5	NA
102	0240	17 142	-60 037	180	5	SA
102	0338	17 084	-60 041	180	5	SA
102	0530	16 575	-60 057	180	5	SA
102	0600			177	5	TP
102	0718	16 471	-60 078	177	5	SA
102	0756	16 432	-60 084	177	5	SA
102	0830			090	5	TP
102	0912	16 393	-60 070	090	5	SA
102	1104	16 393	-59 586	090	5	SA
102	1242	16 386	-59 520	090	5	SA
102	1338	16 381	-59 477	090	5	SA
102	1426	16 382	-59 438	090	5	SA
102	1440			083	5	SA
102	1524	16 386	-59 394	083	5	SA
102	1616	16 408	-59 339	083	5	SA
102	1654	16 403	-59 307	083	5	SA
102	1730			086	5	TP
102	1730			086	5	NA
102	1756	16 1756	-59 247	086	5	SA
102	1842	16 427	-59 204	086	5	SA
102	1915			090	5	TP
102	1940	16 428	-59 141	090	5	SA
102	2030	16 428	-59 094	090	5	SA
102	2042	16 430	-59 081	090	5	SA
102	2215			094	5	TP
102	2224	16 437	-58 581	094	5	SA
103	0106	16 431	-58 426	094	5	SA
103	0115			311	16	TP
103	0250	16 585	-58 599	311	16	SA
103	0334	17 061	-59 085	311	16	SA
103	0438	17 172	-59 210	311	16	SA
103	0522	17 238	-59 303	311	16	SA
103	0545			180	5	TP
103	0628	17 234	-59 348	180	5	SA
103	0818	17 119	-59 355	180	5	SA
103	0830			177	5	TP
103	0854	17 069	-59 357	177	5	SA
103	1010	16 591	-59 363	177	5	SA
103	1130			092	5	TP
103	1256	16 512	-59 284	092	5	SA
103	1328	16 507	-59 251	092	5	SA
103	1436	16 507	-59 186	092	5	SA
103	1516	16 506	-59 145	092	5	SA
103	1626	16 507	-59 079	092	5	SA
103	1645			179	6	TP
103	1725			000	65	TP
103	1752	16 490	-59 065	000	65	SA
103	1815			179	5	TP
103	1848	16 510	-59 060	179	5	SA
103	1940	16 455	-59 060	179	5	SA
103	2040	16 394	-59 067	179	5	SA
103	2134	16 333	-59 069	179	5	SA
103	2145					
9999999						
9999999						

Figure 2. A listing of information input into the automated version of the algorithm that resulted in a generation of optimal track. The plot is shown in figure 1. The two-character designators have the meanings: satellite fix (SA); turning point (TP); and new acceleration parameter (NA). Cards of nine separating solution sets are presented. A detailed discussion is provided in the appendix.

APPENDIX

Introduction

This appendix is designed not only to aid an individual wishing to use the programs written to implement the algorithm, but also to provide further insight into the way in which the solution was set up so that the user can extract the best solution from his data.

We provide two subroutines: SMPLT and DRAWL. The subroutine SMPLT reads in navigation fixes, as in figure 2, and calculates a best-estimate dead-reckoned (DR) track from the ordered courses and speeds. It does this by computing the correction parameter vector $\underline{B}$ for both the x- and y-components of the ship's position vector and then by generating adjusted positions interspersed with the original fixes on an output file having the same format as the input file. The subroutine DRAWL will read the output file of SMPLT and produce a set of charts showing the adjusted track and the fixes used in the adjustment, as in figure 1.

The subroutines are written in FORTRAN, as defined by the American National Standards Institute (Standard X3.9--1966), to make them as machine-independent as possible. Those included have been run on the Univac 1108 and CDC 6600 computers. Nonstandard versions which run on IBM 1130 and IBM 1800 are available.

Smooth Plotting Program Description

This routine, the core of the solution deck, is called into operation as SMPLT (IDT, LUF, LCP, LCR, LLP, NDFV, NDFA). These calling arguments have the following meaning:

IDT	Time interval, in minutes, at which points on the smoothed trackline are generated. For example, in IDT = 5, positions are generated at 5-min intervals and written onto the navigation output file.
LUF	The logical unit designation of the scratch file upon which the fixes are written. Eight words are written on this file for every fix in a data set.
LCP	The logical unit designation of the navigation output file (card punch).
LCR	The logical unit designation of the input file (card reader).
LLP	The logical unit designation of the print file upon which is written the input data, the summary of corrections, and the residuals (line printer).
NDFV	The number of fixes required on a correction line before solving for a constant correction velocity. If there are fewer than NDFV-fixes on a correction line, the ordered course and speed as read from the input file are accepted for that line, and only the position of the line is allowed to change.
NDFA	The number of fixes required on a correction line before an acceleration parameter is added to the solution for that line.

This routine requires a data deck which, aside from spacers and end cards, is formatted (1X, I3, F2.0, F3.1, 2X, F3.0, 1X, F4.2, 1X, F4.0, 1X, F4.2, 1X, F4.1, 2X, F3.1, 1X, A2, 1X, F6.2) with decimal points implied. This will hereafter be described as the Marine Geology

& Geophysics (MG&G) Standard Navigation Format. The fields of this card, an example of which is seen in figure 2, have the following definitions:

Day (I3): contains the Julian Date (JD)

Time (F2.0, F3.1): contains the Greenwich Mean Time in hours and in minutes to tenths of minutes.

Latitude (F3.0, 1X, F4.2) and

Longitude (F4.0, 1X, F4.2): are fix latitude and longitude in degrees and in minutes to hundredths of minutes. The sign convention used is that North latitude and East longitude are positive.

Course (F4.1): is the ordered or estimated course, in degrees to tenths of a degree, measured east from north.

Speed (F3.1): is the ordered or estimated speed, in knots to tenths of a knot.

Type of fix (A2): is a two-character symbol denoting the type of fix; or, if the card does not represent a fix, other control information as described below.

Eötvös correction (F6.2): is the correction, in milligals to hundredths of a milligal, to be applied to gravity observations made on a moving platform. This member is not read off the input cards, but is generated within the subroutine and written on the output card images.

The data deck formatted in this way consists of two types of cards: fix and control. The fix cards must bear the time of fix, the latitude and longitude, an estimated course and speed, and a designator describing the type of fix. Recognized fix labels are: satellite (SA);

Loran A (LA); Loran C (LC); and Omega (OM). These designators are used in a relative weighting of the fixes. Weighting is vested in a data statement within SMPLT and is now set uniformly to one.

Control cards, that is, cards not bearing fix information, are of several types. The first and last cards of a connected track segment are special cards: the first card bears the time of the start of the line as well as ordered course and speed; and the last card of the same set bears only the time of line termination.

A second type of control card is that marking the turning points. In actuality, this type of card is needed only when a turning point and a fix are noncoincident in time. The subroutine sets new correction parameters upon recognition of new speeds or course changes greater than 10° in the input stream. We denote points of velocity change that are not fixes with a TP (turning point), and the record carries only time, course, and speed. Should the latitude or longitude columns bear information, the record will be treated as a fix.

The third possible control card bears only time, course, speed, and NS or NA. The difference between this and the TP control card is solely in the designator. These control cards force a new correction parameter to become active (either new velocity or new acceleration) in the calculation, although the overall course has not been altered. By using only NA cards, one guarantees velocity continuity. These cards can be used to increase the degree of fit by providing additional flexibility for the trackline.

In summary, the following rules must be observed in setting up the data deck:

(1) All fix and control cards must contain a day and time.

(2) Latitude and longitude fields are nonzero only if the card represents a fix.

(3) New speeds and changes of course greater than 10° signal a new set of correction parameters into operation.

(4) NA and NS cards can be used to increase the number of parameters otherwise determined internally.

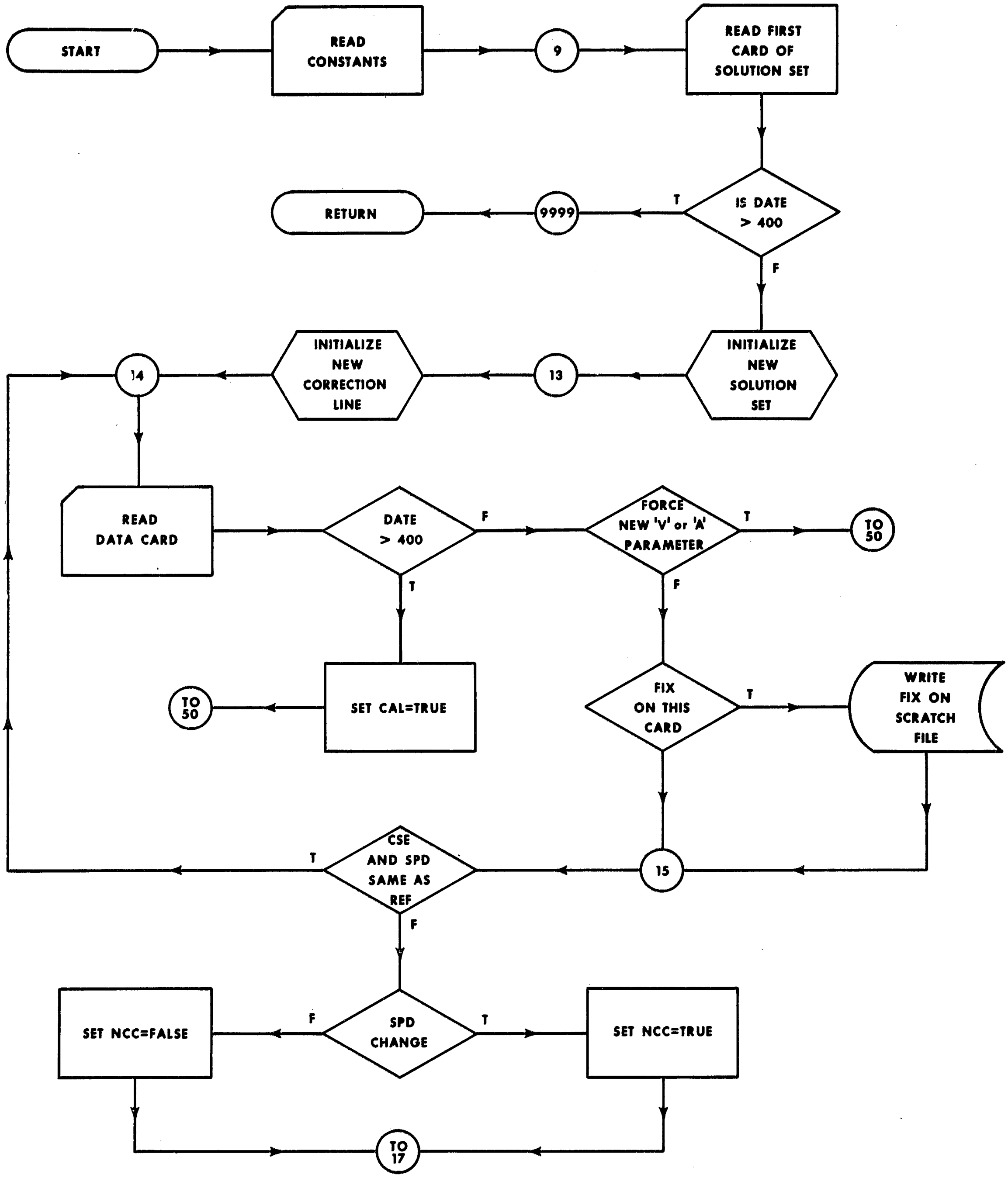
(5) The last card of a data set contains information in the time column only.

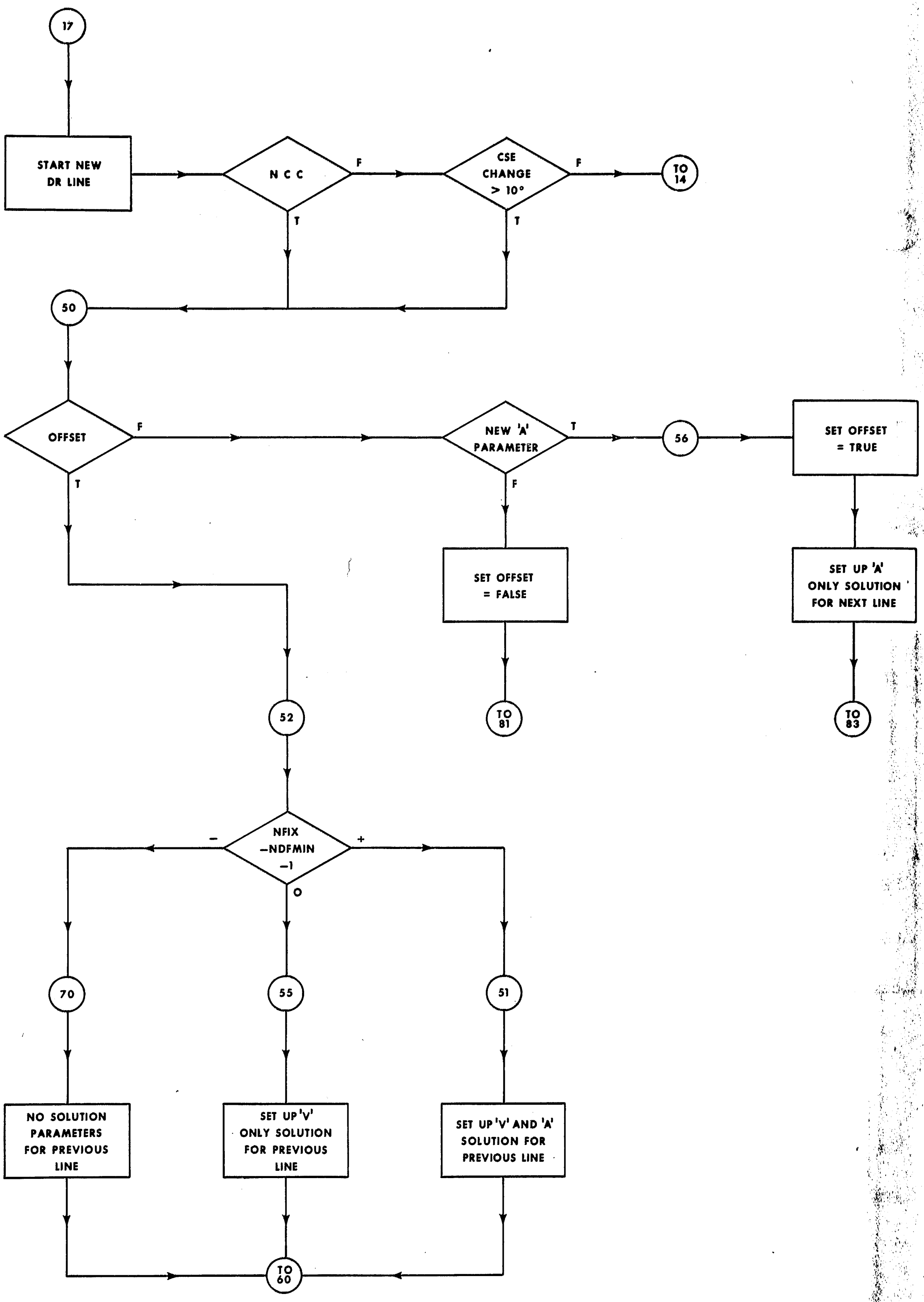
(6) Speeds and courses on sparse track segments, where sparse means the number of fixes is less than NDFV, are accepted at face value and fix the length and angular orientation of that track segment.

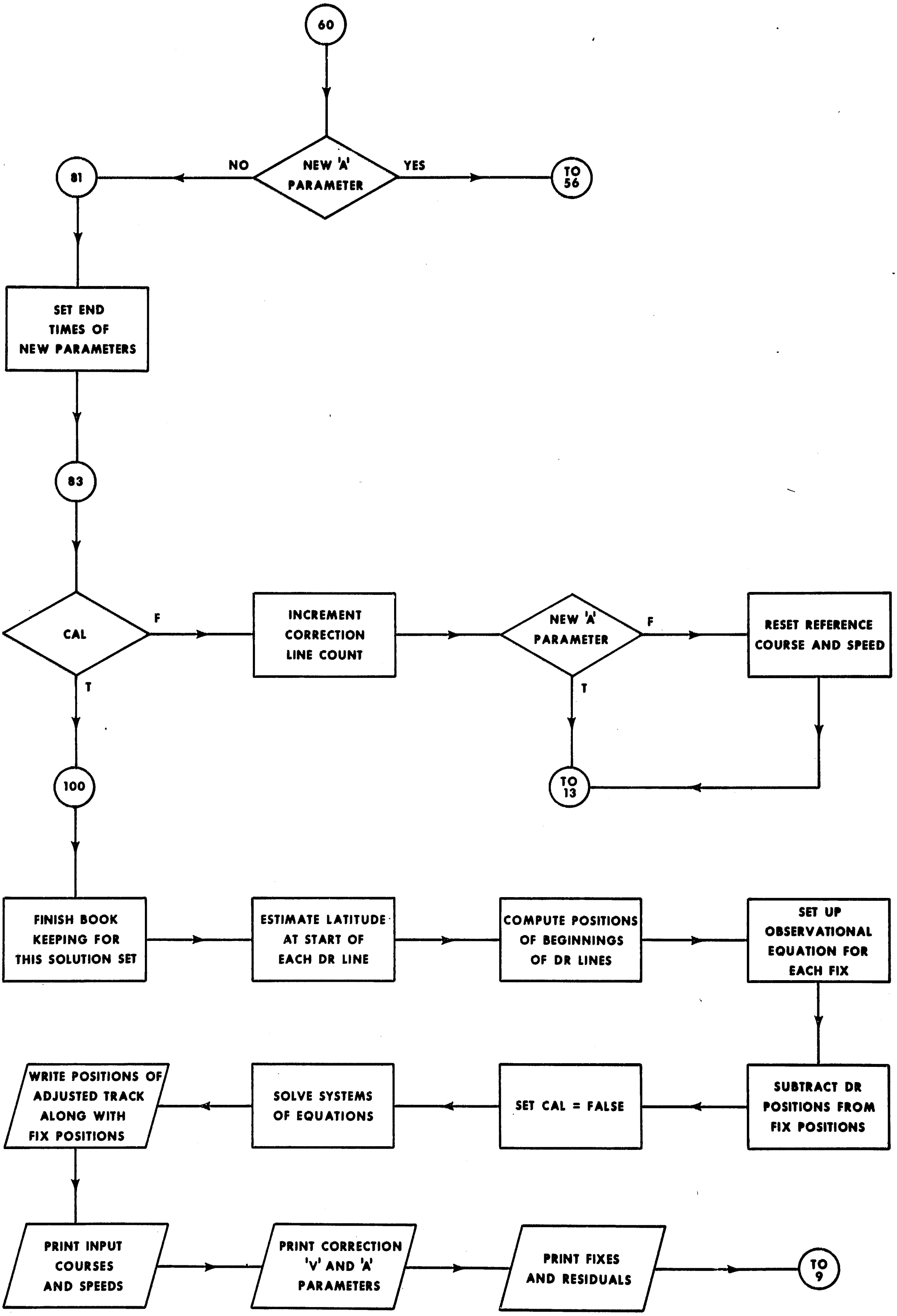
Subroutine SMPLT has both print and type output. The tape output consists of information written in the Standard Navigation MG&G Format, containing every fix read in plus all calculated positions that are designated with a DR. The designators TP, NS, or NA will not appear, although the two-character UP is possible, meaning a break in the trackline. All other information on an UP record is meaningless.

Printed output consists of: an input card list; a summary of correction parameters (i.e., their magnitudes and directions as well as their standard deviations); and lastly, a listing of all fixes and their residuals from the calculated position north and east. This summary is repeated for every solution set.

SUBROUTINE SMPLT




17
START NEW DR LINE
N C C
F
T
CSE CHANGE > 10°
F
T
TO 14
50
OFFSET
F
T
NEW 'A' PARAMETER
T
F
56
SET OFFSET = TRUE
SET OFFSET = FALSE
SET UP 'A' ONLY SOLUTION FOR NEXT LINE
TO 81
TO 83
52
NFIX −NDFMIN −1
−
+
0
70
55
51
NO SOLUTION PARAMETERS FOR PREVIOUS LINE
SET UP 'V' ONLY SOLUTION FOR PREVIOUS LINE
SET UP 'V' AND 'A' SOLUTION FOR PREVIOUS LINE
TO 60


60
NEW 'A' PARAMETER
NO
YES
81
TO 56
SET END TIMES OF NEW PARAMETERS
83
CAL
F
T
INCREMENT CORRECTION LINE COUNT
NEW 'A' PARAMETER
F
T
RESET REFERENCE COURSE AND SPEED
100
TO 13
FINISH BOOK KEEPING FOR THIS SOLUTION SET
ESTIMATE LATITUDE AT START OF EACH DR LINE
COMPUTE POSITIONS OF BEGINNINGS OF DR LINES
SET UP OBSERVATIONAL EQUATION FOR EACH FIX
SUBTRACT DR POSITIONS FROM FIX POSITIONS
SET CAL = FALSE
SOLVE SYSTEMS OF EQUATIONS
WRITE POSITIONS OF ADJUSTED TRACK ALONG WITH FIX POSITIONS
PRINT INPUT COURSES AND SPEEDS
PRINT CORRECTION 'V' AND 'A' PARAMETERS
PRINT FIXES AND RESIDUALS
TO 9

A complete indexed listing of the subroutine is provided herein. A detailed flow chart will guide the interested programer through the bookkeeping maze integral to SMPLT.

Display Program Description

This display program, designed to allow visual integration of the data and solution, is called into operation as DRAWL (ALAT, ALONG, NX, NY, A, IDELT, NIN, NOUT, NSCAT, NPLOT) where:

ALAT ALONG	One-dimensional arrays representing the limiting latitudes and longitudes of a sequence of Mercator plots requisite to the display of a trackline and navigation fixes over a given area. The content of each array must be arranged in ascending sequence (west of Greenwich negative) and must represent whole or one-half degrees only in decimal degrees.
NX NY	Dimension of ALAT and ALONG, respectively. Each integer must be between two and ten.
A	Plot scale in inches/degree of longitude.
IDELT	Time increment in minutes at which time ticks will be made (the routine checks the input file for integral multiples of this unit).
NIN	The logical unit designation of the input file. This BCD-coded file will correspond to the output file (LCP) of the subroutine SMPLT.
NOUT	The logical unit designation of the printed output file.

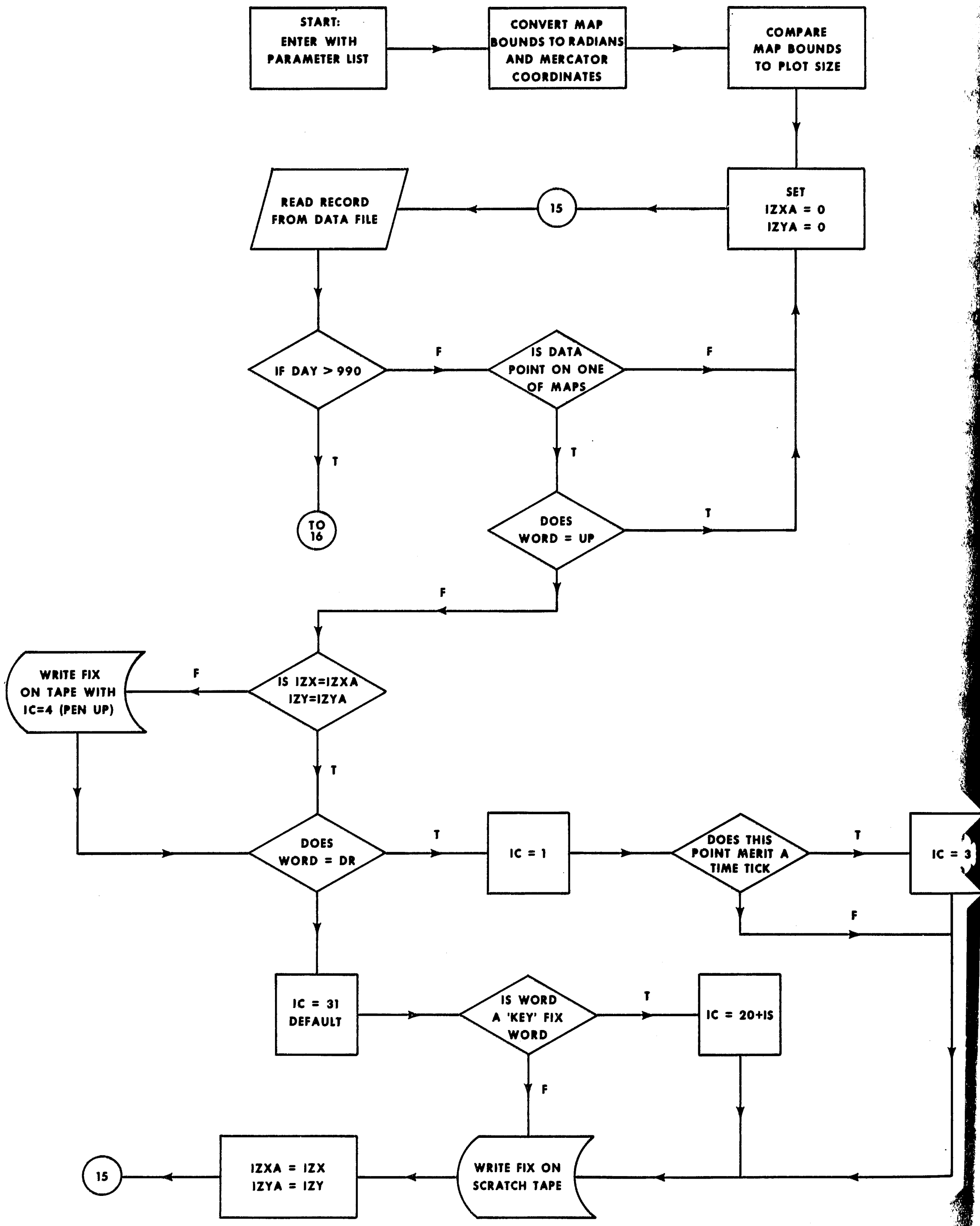
NSCAT	The logical unit description of a scratch file.
NPLOT	The logical unit designation of the plot tape file.

This subroutine allows one to plot both fixes as well as the trackline, as in figure 1, on a sequence of Mercator charts which, when manually spliced, serve to describe an area of any size at any scale. Capability to cross the trackline at set time intervals is also provided. The routine requires only a delineation of latitude and longitude map bounds and an input file (generally the output file of SMPLT), containing a time-ordered sequence of points defined by a time, a position, and a two-character descriptor of the point's genesis. The input file is BCD-coded in the MG&G Standard Navigation Format which has been described above.

The two-character descriptor may be one of the following: a point lying on a trackline (DR); a satellite fix (SA); an Omega fix (OM); a Loran A fix (LA); a Loran C fix (LC); and a break in the trackline (UP). When the DR is identified in the input file, a check is made to see if the accompanying time is a multiple of the time-tick increment, allowing the DR line to be crossmarked. All other designators, aside from the UP, will be plotted with an appropriate symbol. Adjustments to the list of recognized fixes can be accomplished with minor changes to data statements.

The subroutine will sequentially plot up to 81 Mercator charts of a contiguous area. Control of the plot boundaries is maintained through calling parameters, and a sorting of points and trackline to individual charts is done efficiently within the routine. Graticules,

SUBROUTINE DRAWL



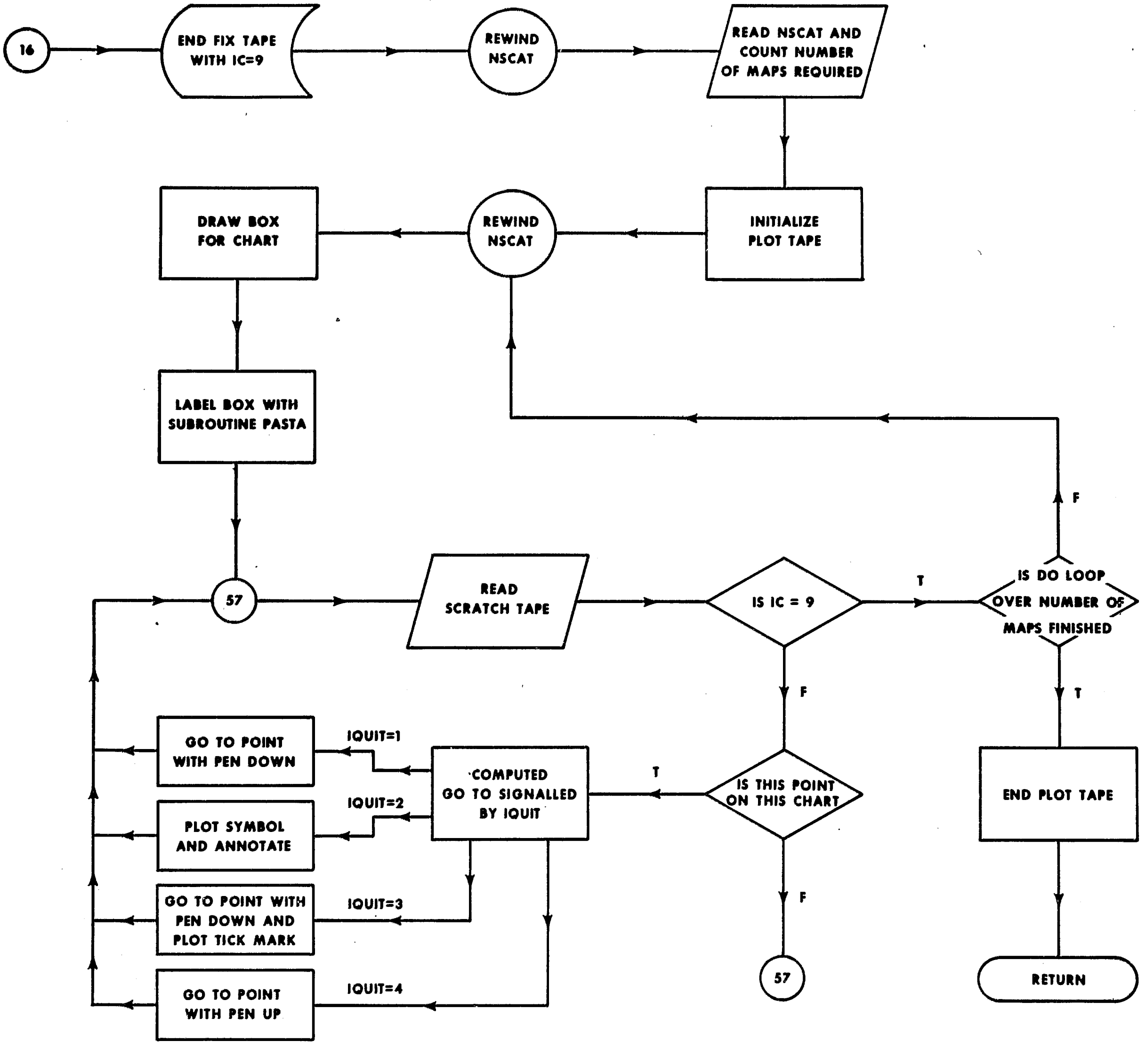

16
END FIX TAPE WITH IC=9
REWIND NSCAT
READ NSCAT AND COUNT NUMBER OF MAPS REQUIRED
INITIALIZE PLOT TAPE
REWIND NSCAT
DRAW BOX FOR CHART
LABEL BOX WITH SUBROUTINE PASTA
57
READ SCRATCH TAPE
IS IC = 9
T
IS DO LOOP OVER NUMBER OF MAPS FINISHED
F
T
END PLOT TAPE
RETURN
F
IS THIS POINT ON THIS CHART
T
F
57
COMPUTED GO TO SIGNALLED BY IQUIT
IQUIT=1
GO TO POINT WITH PEN DOWN
IQUIT=2
PLOT SYMBOL AND ANNOTATE
IQUIT=3
GO TO POINT WITH PEN DOWN AND PLOT TICK MARK
IQUIT=4
GO TO POINT WITH PEN UP

chart labeling, and plot-drift indicators are drawn for each chart. The plot-tape output is initialized and terminated within the subroutine. Control of this aspect can be recovered by the programer by deactivating statements DRW 1300 and DRW 2260.

Output consists of a CalComp command tape which will create a diagram, as in figure 1, as well as a printed output review of the subroutine argument assignments. A complete indexed listing of the subroutine is provided herein as well as a detailed flow chart.

Other Considerations

A complete listing of all nonstandard subroutines, called by SMPLT and DRAWL, has been provided. Comment cards in each subroutine describe their function as well as the meanings of the calling arguments. In addition to the listed routines, calls are made to two IBM standard matrix-manipulative routines, MINV and GMPRD (IBM, 1970), as well as to three CalComp standard routines, PLOT, SYMBOL, and NUMBER.

Also included in the listings are two driver programs, called DRI and DRA, which will indicate the way we have employed SMPLT and DRAWL. We have run each separately because of the insufficient core available in our general-use computer. The user may wish to follow this example or create a single overall driver program if a larger machine is available.

Notation used in SMPLT is identical to that used in the description of the algorithm. That fact, along with the accompanying flow chart, should provide the interested reader with sufficient material to guide him through the program's internal intricacies. The flow chart and previous description of DRAWL should provide the same service.

We believe that we have developed a useful tool for geophysical navigation work. We therefore hope that these descriptions will provide sufficient insight into the machinery of the automated algorithm that the reader will be able to make as much use of the routines as we think they merit.

Reference

IBM (1970): *System/360 Scientific Subroutine Package Programmer's Manual*, IBM Corp., Technical Publications Dept.

Program Listings

```
      DATA LCR/5/
      READ (LCR,708) IDT,LUF,LCP,LCR,LLP,NDFV,NDFA
  708 FORMAT (7I5)
      CALL SMPLT(IDT,LUF,LCP,LCR,LLP,NDFV,NDFA)
      END
```

```
      SUBROUTINE SMPLT(IDT,LUF,LCP,LCR,LLP,NDFV,NDFA)
      LOGICAL CAL,OFFSET,NCC
      DIMENSION A(500,50),AI(60,60),AX(50),AY(50),B(500),C(500),CS(50),
     1FLA(50),IT(60),PARX(60),PARY(60),SDPX(60),SDPY(60),SP(50),
     2T(50),TB(60),TE(60),W(500),X(500),Y(500),IAL(3)
      DATA NUP/2HUP/,IAL(2),IAL(3)/4H    ,4H/HR./
      DATA RAD/.01745329252/,IDR,NA,NS/2HDR,2HMA,2HNS/
C FAC IS CM/HR/KT AT THE EQUATOR
C EPS MUST BE SUCH THAT 9000. IS RESOLVABLE FROM 9000.+EPS
C DDR IS RHE TIME,EXPRESSED IN HOURS, BETWEEN THE GENERATED POSITIONS
      DATA FAC/0.3386673/,EPS/1.0E-4/,DDR/0.08333333333/
      DATA NRA,NCA,NLMX/500,60,50/
CONVERT ANGLES TO DEGREES
      ANG(P1,P2)=P1    +SIGN(P2,P1)/60.
CONVERT TIMES TO HOURS FROM BEGINNING OR YEAR
      TIME(P1,P2,P3)=24.*P1+P2+P3/60.
COMPUTE EOTVOS CORRECTION AS A FUNCTION OF COURSE,SPEEDIN KNOTS,LATITUDE
C IN DEGREES
      EOTVOS(P1,P2,P3)=7.50277*P2*SIN(P1*RAD)*COS(P3*RAD)+.0004149*P2*P2
   71 FORMAT (1X,F3.0,1X,F2.0,F3.1,2X,F3.0,1X,F4.2,1X,F4.0,1X,F4.2,1X,
     * F4.1,2X,F3.1,1X,A2)
   72 FORMAT(I5,2F12.0)
   73 FORMAT(1X,F4.0,2(1X,F3.0),2(F5.0,F6.2),2F6.1,1X,A2)
   74 FORMAT(10F12.2)
   75 FORMAT(10I12)
   76 FORMAT (////////// 50X,12HNORMAL EXIT.)
   77 FORMAT (10E12.3)
  701 FORMAT (1H1,48X,24HINPUT COURSES AND SPEEDS,/, 1H0,41X,4HFROM, 9X,
     12HTO,   7X,6HCOURSE, 4X,5HSPEED,/,1H ,37X,2(13HDAY  TIME    ),3X,
     2 7HDEGREES, 3X,5HKNOTS )
  702 FORMAT (38X,2(I3,F6.2,4X),2(4X,F5.1,5X))
  703 FORMAT(////30X,47HCORRECTIONS APPLIED TO INPUT COURSES AND SPEEDS
     1 /1H0,36X,4HFROM,9X,2HTO,41X,4HSTD./32X,2(13HDAY  TIME    ),6X,
     2 5HNORTH,14X,4HEAST,5X,4HDEV.)
  704 FORMAT (33X,2(I3,F6.2,4X),2(F5.2,6H KNOTS,A4,3X),2F5.2)
  705 FORMAT (////50X,15HCONTROL SUMMARY//1X,64H LINE   DAY  TIME   LATI
     1TUDE  LONGITUDE   ADJUSTMENT MADE   TYPE /19X,44HDEGREES N DEGREES
     2 E   NAUTICAL MILES    FIX  /43X,13HNORTH    EAST /)
  706 FORMAT (I4,I7,F6.2,2X,2F10.3,2F8.2,6X,A2)
  707 FORMAT (10X,40HTHE DETERMINANT OF THE NORMAL MATRIX IS   ,E10.3)
  709 FORMAT (//10X,56HSMPLTR WAS CALLED WITH THE FOLLOWING FILE CONFIGU
     *RATION /10X,5HINPUT,15X,I4/10X, 7HSCRATCH,13X,I4/10X, 17HNAVIGATIO
     *N OUTPUT,3X,I4,/10X, 12HPRINT OUTPUT,8X,I4,//10X,I2,  75H FIXES RE
     *QUIRED ON A CORRECTION LINE BEFORE A VELOCITY SOLUTION IS ALLOWED.
     */  10X,I2,81H FIXES REQUIRED ON A CORRECTION LINE BEFORE AN ACCELE
     *RATION SOLUTION IS ALLOWED. //10X, 45HSMOOTHED OUTPUT POINTS ARE T
     *O BE GENERATED AT,I4,17H MINUTE INTERVALS ////)
  708 FORMAT(1H1)
      WRITE (LLP,709) LCR,LUF,LCP,LLP,NDFV,NDFA,IDT
      DDR=FLOAT(IDT)/60.0
      CALL MERC (30.,-80.,8.0,0.0,1)
    9 CONTINUE
      WRITE(LLP,708)
      CAL=.FALSE.
      OFFSET=.FALSE.
C SET UP NEW SOLUTION
      REWIND LUF
      NEQ=0
      DO 10 I=1,NCA
      DO 10 J=1,NRA
```

```
   10 A(J,I)=0.0                                                        SMP  610
      DO 11 I=1,NLMX                                                    SMP  620
      IT(I)=0                                                           SMP  630
      TB(I)=0.0                                                         SMP  640
      TE(I)=0.0                                                         SMP  650
   11 CONTINUE                                                          SMP  660
      DO 12 I=1,NRA                                                     SMP  670
      B(I)=0.0                                                          SMP  680
      C(I)=0.0                                                          SMP  690
      A(I,1) =1.0                                                       SMP  700
   12 CONTINUE                                                          SMP  710
      IXB=2                                                             SMP  720
      I=1                                                               SMP  730
      LN=1                                                              SMP  740
      L=0                                                               SMP  750
      IT(1)=1                                                           SMP  760
      READ(LCR,71)FD,FH,FM,FAD,FAM,FOD,FOM,CSE,SPD,IFX                  SMP  770
      IF (FD .GT. 400.) GO TO 9999                                      SMP  780
      CSE=CSE*RAD                                                       SMP  790
      TT=TIME(FD,FH,FM)                                                 SMP  800
      T(1)=TT                                                           SMP  810
      TB(1)=TT                                                          SMP  820
      TB(2)=TT                                                          SMP  830
      TB(3)=TT                                                          SMP  840
      TR=TT                                                             SMP  850
      CS(1)=CSE                                                         SMP  860
      SP(1)=SPD                                                         SMP  870
      RCSE=CSE                                                          SMP  880
      RSPD=SPD                                                          SMP  890
C SET UP NEW LINE                                                       SMP  900
   13 NFIX =0                                                           SMP  910
C READ A CARD                                                           SMP  920
   14 READ(LCR,71)FD,FH,FM,FAD,FAM,FOD,FOM,CSE,SPD,IFX                  SMP  930
      WRITE (LLP,73) FD,FH,FM,FAD,FAM,FOD,FOM,CSE,SPD,IFX               SMP  940
      IF (FD .LT. 400.) GO TO 141                                       SMP  950
      CAL=.TRUE.                                                        SMP  960
      GO TO 50                                                          SMP  970
  141 TT=TIME(FD,FH,FM)                                                 SMP  980
      CSE=CSE*RAD                                                       SMP  990
      FA=ANG(FAD,FAM)                                                   SMP 1000
      FO=ANG(FOD,FOM)                                                   SMP 1010
      IF (IFX .EQ. NS) GO TO 50                                         SMP 1020
      IF (IFX .EQ. NA) GO TO 50                                         SMP 1030
      IF(FA.EQ.0.0.AND.FO.EQ.0.0) GO TO 15                              SMP 1040
      NEQ=NEQ+1                                                         SMP 1050
      NFIX=NFIX+1                                                       SMP 1060
      WRITE(LUF)TT,FA,FO,CSE,SPD,IFX,NEQ,LN                             SMP 1070
   15 IF (((SPD-RSPD)**2+(CSE-RCSE)**2) .LT. EPS) GO TO 14              SMP 1080
      NCC=.FALSE.                                                       SMP 1090
      IF (SPD .NE. RSPD) NCC=.TRUE.                                     SMP 1100
   17 CONTINUE                                                          SMP 1110
      LN=LN+1                                                           SMP 1120
      T(LN)=TT                                                          SMP 1130
      CS(LN)=CSE                                                        SMP 1140
      SP(LN)=SPD                                                        SMP 1150
      IF (NCC) GO TO 50                                                 SMP 1160
      DA=COS(CSE)*COS(RCSE)+SIN(CSE)*SIN(RCSE)                          SMP 1170
      RCSE=CSE                                                          SMP 1180
      RSPD=SPD                                                          SMP 1190
      IF ( DA .LT. .985) GO TO 50                                       SMP 1200
```

```
      GO TO 14
C SET UP NEW CORRECTION COURSE
   50 WRITE (LLP,72) NFIX,TT
      IF (.NOT. OFFSET) GO TO 52
      IF (IFX.EQ. NA) GO TO 56
      OFFSET=.FALSE.
      GO TO 81
   52 CONTINUE
      IF (NFIX .GE. NDFA) GO TO 51
      IF (NFIX .GE. NDFV) GO TO 55
      GO TO 70
C V AND A SOLUTION
   51 IQ=3
      TB(IXB)=TR
      TB(IXB+1)=TR
      TR=TT
      IT(IXB)=2
      IT(IXB+1)=3
      IXB=IXB+2
      GO TO 60
C V ONLY SOLUTION
   55 IQ=2
      TB(IXB)=TR
      TR=TT
      IT(IXB)=2
      IXB=IXB+1
      GO TO 60
   70 IQ=1
C NO SOLUTION PARAMETERS ON THIS LINE
      TR=TT
      GO TO 60
C A ONLY SOLUTION
   56 IF (OFFSET) GO TO 57
      OFFSET=.TRUE.
   57 IT(IXB)=3
      TB(IXB)=TT
      IXB=IXB+1
      GO TO 81
   60 IF (IFX .EQ. NA) GO TO 56
   81 IF (IFX .EQ. NA) GO TO 83
      DO 82 J=2,IXB
      IF(TE(J-1) .EQ. 0.0) TE(J-1)=TT
   82 CONTINUE
   83 CONTINUE
      IF (CAL) GO TO 100
      L=L+1
      IF (IFX .EQ. NA) GO TO 13
      RCSE=CSE
      RSPD=SPD
      TR=TT
      GO TO 13
  100 CONTINUE
CALCULATE SOLUTION
      NPAR=IXB-1
      NMX=NEQ
      NLN=LN
C ESTIMATE LAT AT BEGINNING PF EACH DR LINE BY LINEAR INTERPOLATION IN
C FIX LATITUDES
      NEQ=0
      REWIND LUF
```

```
      T2=0.
      F2=0.
      DO 510 I=1,NLN
  505 IF (T(I) .LE. T2 .OR. NEQ .EQ. NMX) GO TO 510
      T1=T2
      F1=F2
      READ (LUF) T2,F2,FO,CSE,SPD,IFX,NEQ,IX
      GO TO 505
  510 FLA(I)=F1+(F2-F1)*(T (I)-T1)/(T2-T1)
COMPUTE POSITIONS FOR BEGINNINGS OF EACH DR LINE
      AX(1)=0.0
      AY(1)=0.0
      DO 150 I=2,NLN
      I1=I-1
      TT=(T(I)-T(I1))*FAC*SP(I1)/COS(FLA(I1)*RAD)
      AX(I)=AX(I1)+TT*SIN(CS(I1))
      AY(I)=AY(I1)+TT*COS(CS(I1))
  150 CONTINUE
C     PRINT 74,(CS(I),SP(I),I=1,NLN)
C     PRINT 74,(AX(I),AY(I),I=1,NLN)
C     PRINT 74,(TB(I),TE(I),I=1,NPAR)
C     PRINT 74,(T(I),I=1,NLN)
C     PRINT 74,(FLA(I),I=1,NLN)
      REWIND LUF
      DO 200 J=1,NMX
      READ(LUF)TT,FA,FO,CSE,SPD,IFX,NEQ,LN
C WRITE OBSERVATIONAL EQUATIONS FOR FIX
C ASSIGN WEIGHT TO FIX
      W(NEQ)=1.0
      DO 215 I=1,NPAR
      IS=IT(I)
      GO TO (101,102,102),IS
  101 A(NEQ,I)=1.0
      GO TO 110
  102 IF (TT .LT. TB(I)) GO TO 110
      TTT=AMIN1(TT,TE(I))-TB(I)
      GO TO (110,103,104),IS
  103 A(NEQ,I)=TTT
      GO TO 110
  104 A(NEQ,I)=TTT**2*0.5
  110 CONTINUE
  215 CONTINUE
C DIFFERENCE FIX AND DR POSITIONS
      CALL MERC(FA,FO,TX,TY,2)
      TTT=(TT-T(LN))/(T(LN+1)-T(LN))
      PX=AX(LN)+(AX(LN+1)-AX(LN))*TTT
      PY=AY(LN)+(AY(LN+1)-AY(LN))*TTT
      B(NEQ)=TX-PX
      C(NEQ)=TY-PY
  200 CONTINUE
      REWIND LUF
      CAL=.FALSE.
C     DO 900 I=1,NMX
C 900 PRINT 74,     (A(I,J),J=1,NPAR)
      CALL LINLSQ(A,NRA,NEQ,NPAR,B,W,PARX,     X,SDPX,AI,DET)
      CALL LINLSQ(A,NRA,NEQ,NPAR,C,W,PARY,     Y,SDPY,AI,DET)
C     PRINT 74,(B(I),C(I),I=1,NMX)
C     PRINT 74,(PARX(I),PARY(I),I=1,NPAR)
C     PRINT 74,(X(I),Y(I),I=1,NMX)
C     PRINT 74,(SDPX(I),SDPY(I),I=1,NPAR),DET
```

```
CALCULATE DR POSITIONS AND WRITE ALONG WITH FIXES.
      TFRST=T(1)
      TLST=T(NLN)
      LN=0
      TP=TFRST+EPS
      CAL=.FALSE.
      IX=TFRST/DDR
 1002 READ (LUF) TT,FA,FO,CSE,SPD,IFX,NEQ,L
      CSE=CSE/RAD
 1005 IF(TP-TT) 1015,1010,1010
C WRITE FIX
 1010 ETV=EOTVOS(CSE,SPD,FA)
      CALL PNAVC(TT,FA,FO,CSE,SPD,IFX,ETV,LCP)
      IF (NEQ-NMX) 1002,1014,1014
 1014 TT=9999.
COMPUTE DR POSITION
CHECK TO SEE THAT WE ARE ON HE  CORRECT DR LINE
 1015 IF (TP-T(LN+1)) 1017,1017,1016
 1016 LN=LN+1
      TP=TP+EPS
      DT=T(LN+1)-T(LN)
      DX=AX(LN+1)-AX(LN)
      DY=AY(LN+1)-AY(LN)
C VLO IS LOCAL VELOCITY SCALE IN CM/HR/KT
      VLO=FAC/COS(FLA(LN)*RAD)
      VXB=VLO*SP(LN)*SIN(CS(LN))
      VYB=VLO*SP(LN)*COS(CS(LN))
      IF(ABS(TP-DDR/2.-T(LN))-.45*DDR) 1020,1020,1017
C SET UP TO PUNCH POSITION OF LINE CORNER
 1020 IX=IX-1
      TP=T(LN)
C INTERPOLATE ON UNCORRECTED DR LINE
 1017 TTT=(TP-T(LN))/DT
      PX=AX(LN)+TTT*DX
      PY=AY(LN)+TTT*DY
      VX=VXB
      VY=VYB
C PX,PY ARE UNCORRECTED DR POSITIONS IN MAP COORDINATES
C LOOP THRU CORREDTION PARAMETERS AND SUM TO GET CORRECTED POSITION
      DO 1110 I=1,NPAR
      IS=IT(I)
      GO TO (1101,1102,1102),IS
 1101 PX=PX+PARX(I)
      PY=PY+PARY(I)
      GO TO 1110
 1102 IF (TP .LT. TB(I)) GO TO 1215
      TTT=AMIN1(TP,TE(I))-TB(I)
      GO TO (1101,1103,1104).IS
 1103 PX=PX+TTT*PARX(I)
      PY=PY+TTT*PARY(I)
      VX=VX+PARX(I)
      VY=VY+PARY(I)
      GO TO 1110
 1104 VX=VX+TTT*PARX(I)
      VY=VY+TTT*PARY(I)
      TTT=TTT**2*0.5
      PX=PX+TTT*PARX(I)
      PY=PY+TTT*PARY(I)
 1110 CONTINUE
 1215 CONTINUE
```

```
      TSPD=VX*VX+VY*VY                                                  SMP 3010
      IF(TSPD) 1216,1216,1217                                           SMP 3020
 1216 TSPD=0.0                                                          SMP 3030
      TCSE=0.0                                                          SMP 3040
      GO TO 1218                                                        SMP 3050
 1217 TSPD=SQRT(TSPD)/VLO                                               SMP 3060
      TCSE=ATAN2(VX,VY)/RAD                                             SMP 3070
      IF (TCSE) 1219,1218,1218                                          SMP 3080
 1219 TCSE=TCSE+360.                                                    SMP 3090
 1218 CONTINUE                                                          SMP 3100
CONVERT FROM MAP COORDINATES TO LAT,LON AND WRITE DR POS                SMP 3110
      CALL MERC (FLT,FLO,PX,PY,3)                                       SMP 3120
      ETV=EOTVOS(TCSE,TSPD,FLT)                                         SMP 3130
      CALL PNAVC (TP,FLT,FLO,TCSE,TSPD,IDR,ETV,LCP)                     SMP 3140
      IF (CAL) GO TO 2000                                               SMP 3150
      IX=IX+1                                                           SMP 3160
      TP=FLOAT(IX)*DDR                                                  SMP 3170
      IF(TP-TLST) 1005,1220,1220                                        SMP 3180
 1220 CAL=.TRUE.                                                        SMP 3190
      TP=TLST                                                           SMP 3200
      GO TO 1005                                                        SMP 3210
 2000 CONTINUE                                                          SMP 3220
      CALL PNAVC (TP,FLT,FLO,TCSE,TSPD,NUP,0.0,LCP)                     SMP 3230
C PRINT INPUT COURSES AND SPEEDS                                        SMP 3240
      WRITE (LLP,701)                                                   SMP 3250
      LIM=NLN-1                                                         SMP 3260
      CALL TYME(T(1),IDAY,FH)                                           SMP 3270
      DO 2100 I=1,LIM                                                   SMP 3280
      CALL TYME (T(I+1),JDAY,FJ)                                        SMP 3290
      CSE=CS(I)/RAD                                                     SMP 3300
      WRITE (LLP,702) IDAY,FH,JDAY,FJ,CSE,SP(I)                         SMP 3310
      IDAY=JDAY                                                         SMP 3320
      FH=FJ                                                             SMP 3330
 2100 CONTINUE                                                          SMP 3340
C PRINT CORRECTIONS                                                     SMP 3350
      WRITE (LLP,703)                                                   SMP 3360
      LN=0                                                              SMP 3370
      DO 2200 I=2,NPAR                                                  SMP 3380
      CALL TYME (TB(I),IDAY,FH)                                         SMP 3390
      CALL TYME (TE(I),JDAY,FJ)                                         SMP 3400
 2130 IF (TB(I) .LT. T(LN+1)) GO TO 2150                                SMP 3410
      LN=LN+1                                                           SMP 3420
      VLO=FAC/COS(FLA(LN)*RAD)                                          SMP 3430
      GO TO 2130                                                        SMP 3440
 2150 T1=PARX(I)/VLO                                                    SMP 3450
      T2=PARY(I)/VLO                                                    SMP 3460
      T3=SDPX(I)/VLO                                                    SMP 3470
      T4=SDPY(I)/VLO                                                    SMP 3480
      IX=IT(I)                                                          SMP 3490
      WRITE (LLP,704) IDAY,FH,JDAY,FJ,T2,IAL(IX),T1,IAL(IX),T4,T3       SMP 3500
 2200 CONTINUE                                                          SMP 3510
C PRINT FIXES AND RESIDUALS                                             SMP 3520
      WRITE (LLP,705)                                                   SMP 3530
      REWIND LUF                                                        SMP 3540
      DO 2300 I=1,NMX                                                   SMP 3550
      READ (LUF) TT,FA,FO,CSE,SPD,IFX,NEQ,LN                            SMP 3560
      CSE=CSE/RAD                                                       SMP 3570
      CALL TYME (TT,JDAY,FH)                                            SMP 3580
      VLO=FAC/COS(FLA(LN)*RAD)                                          SMP 3590
      X(NEQ)=X(NEQ)/VLO                                                 SMP 3600
```

```
      Y(NEQ)=Y(NEQ)/VLO                                                 SMP 3610
      WRITE(LLP,706) LN,JDAY,FH,FA,FO,Y(NEQ),X(NEQ),IFX                 SMP 3620
 2300 CONTINUE                                                          SMP 3630
      WRITE (LLP,707) DET                                               SMP 3640
      GO TO 9                                                           SMP 3650
 9999 CONTINUE                                                          SMP 3660
      CALL PNAVC (23976.,0.,0.,0.,0.,NUP,0.0,LCP)                       SMP 3670
      WRITE (LLP,76)                                                    SMP 3680
      RETURN                                                            SMP 3690
      END                                                               SMP 3700
```

```
      SUBROUTINE MERC (A,B,C,D,ICON)
C FLATTENING FOR INTERNATIONAL SPHEROID=1/297
      DATA PIFOR/0.78539816/,RAD/0.0174532925/,EPS/.0819918905/
CALLING SEQUENCE
COMPUTE MERCATOR COORDINATES FROM LAT,LON.
C           CALL MERC(A,B,C,D,ICON)
C DEFINE TRANSFORMATION           ICON=1
C      A,B=LAT,LON IN DEGREES OF ORIGIN OF MERCATOR COORDINATE SYSTEM.
C      C=SCALE IN INCHES/DEGREE
C      D=DUMMY
C      CONVERT COORDINATES    ICON=2
C      A,B=LAT,LON OF POINT
C      C,D=X,Y IN CM. FROM MAP ORIGIN
C TRANSFORM MAP COORDINATES TO LAT,LON
C                                 ICON=3
CALLING PARAMETERS SAME AS FOR ICON=2
      GO TO (1,10,50),ICON
    1 SCALE=C*2.540005
      RSCAL=SCALE/RAD
C SCALE IS IN CM/DEGREE
   10 X=SCALE*B
      RPHI=A*RAD
      EPSIN=SIN(RPHI)*EPS
      Y=RSCAL*ALOG(TAN(PIFOR+RPHI*.5)*((1.-EPSIN)/(1.+EPSIN))**(EPS*.5))
      GO TO (20,30),ICON
   20 XBASE=X
      YBASE=Y
   25 RETURN
   30 C=X-XBASE
      D=Y-YBASE
   35 RETURN
   50 B=(C+XBASE)/SCALE
      FAC=EXP((D+YBASE)/RSCAL)
      EPSIN=EPS*SIN(2.*(ATAN(FAC)-PIFOR))
      E=2.*(ATAN(FAC*((1.+EPSIN)/(1.-EPSIN))**(EPS*.5))-PIFOR)
      EPSIN=EPS*SIN(E)
      A=2.*(ATAN(FAC*((1.+EPSIN)/(1.-EPSIN))**(EPS*.5))-PIFOR)/RAD
      RETURN
      END
```

```
      SUBROUTINE TYME (T,JDAY,FH)                                       TYM   10
CONVERTS TIME T MEASURED IN HOURS FROM THE BEGINNING OF THE YEAR TO     TYM   20
C JULIAN DATE JDAY AND FH WHICH IS HR.MIN                               TYM   30
      DATA ROUND /0.49999999999/                                        TYM   40
      ITEMP=T*60.+ROUND                                                 TYM   50
      JDAY=ITEMP/1440                                                   TYM   60
      ITEMP=ITEMP-1440*JDAY                                             TYM   70
      IH=ITEMP/60                                                       TYM   80
      ITEMP=ITEMP-IH*60                                                 TYM   90
      FH=FLOAT(IH)+FLOAT(ITEMP)/100.                                    TYM  100
      RETURN                                                            TYM  110
      END                                                               TYM  120
```

```
      SUBROUTINE LINLSQ(A,NRA,NOB,NPAR,C,W,B,E,F,AI,DET)
C SOLVES THE MATRIX EQUATION A*B=C WHERE THE SYSTEM IS EVENDETERMINED OR
C OVERDETERMINED---WITH THE OBSERVATIONAL EQUATIONS WEIGHTED BY THE
C VECTOR W.
C NRA IS THE DECLARED ROW DIMENSION OF A
C NOB IS THE NUMBER OF EQUATIONS (THE NUMBER OF ROWS OF A ACTUALLY USED)
C NPAR IS THE DIMENSION OF B AND C (THE NUMBER OF UNKNOWNS)
C E WILL CONTAIN THE RESIDUAL VECTOR
C F WILL CONTAIN THE ESTIMATED STANDARD DEVIATION OF THE PARAMETERS
C AI WILL CONTAIN THE INVERSE OF THE OF THE NORMAL MATRIX
C DET WILL CONTAIN THE DETEERMINANT OF THE NORMAL MATRIX
      DIMENSION A(NRA,NPAR),AI(NPAR,NPAR),B(NOB),C(NOB),W(NOB),E(NOB),
     * F(NPAR)
C WIEGHT
      DATA LOUT/6/
      DO 11 I=1,NOB
      DO 10 J=1,NPAR
   10 A(I,J)=A(I,J)*W(I)
   11 C(I)=C(I)*W(I)
C  IF SYSTEM IS EVENDETERMINED,SKIP TRANSPOSE MULTIPLICATION.
      IF(NOB .EQ. NPAR) GO TO 670
C PREMULTIPLY BY A TRANSPOSE
      DO 550 J=1,NPAR
      DO 550 I=1,NPAR
      SUM=0.0
      DO 525 K=1,NOB
  525 SUM=SUM+A(K,J)*A(K,I)
  550 AI(J,I)=SUM
      GO TO 680
  670 DO 675 I=1,NPAR
      DO 675 J=1,NOB
  675 AI(J,I)=A(J,I)
  680 CONTINUE
C SOLVE SYSTEM
      CALL MINV(AI,NPAR,DET,E,F)
C  F=C*AT
      IF (NOB.NE.NPAR) GO TO 562
      DO 563 J=1,NPAR
  563 F(J)=C(J)
      GO TO 561
  562 DO 650 J=1,NPAR
      SUM=0.0
      DO 625 K=1,NOB
  625 SUM=SUM+A(K,J)*C(K)
  650 F(J)=SUM
C  B=AI*F
  561 CALL GMPRD(AI,F,B,NPAR,NPAR,1)
      SUM=0.0
      DO 700 I=1,NOB
      TEMP=.0
COMPUTE RESIDUALS
      DO 690 J=1,NPAR
  690 TEMP=TEMP+B(J)*A(I,J)
      TEMP=C(I)-TEMP
      SUM=SUM+TEMP*TEMP
      E(I)=TEMP/W(I)
C UNSCALE OBSERVATIONS
      C(I)=C(I)/W(I)
      DO 700 J=1,NPAR
      A(I,J)=A(I,J)/W(I)
```

```
  700 CONTINUE
      IF(NOB .EQ. NPAR) RETURN
      SIGMSQ=SUM/FLOAT(NOB-NPAR)
      DO 705 I=1,NPAR
      FAC=AI(I,I)*SIGMSQ
      TEMP=SQRT(ABS(FAC))
  705 F(I)=SIGN(TEMP,FAC)
      RETURN
   74 FORMAT(5E12.3)
      END
```

```
      SUBROUTINE PNAVC (TIME,FLAT,FLON,CSE,SPD,IT,ETV,LOUT)             PNA   10
C PUNCHES A CARD IM M3+G NAVIGATION FORMAT                               PNA   20
C TIME IS IN HOURS FROM THE BEGINNING OF THE YEAR,FLAT,FLON,CSE IN DEG. PNA   30
C SPD IN KNOTS.                                                         PNA   40
C LM DORMAN   AOML JUN 1973                                             PNA   50
      DATA IB,NM,ROUND/1H ,1H-,0.49999999999/                           PNA   60
      IF(FLAT)2,1,1                                                     PNA   70
    1 ISA=IB                                                            PNA   80
      GO TO3                                                            PNA   90
    2 ISA=NM                                                            PNA  100
    3 CONTINUE                                                          PNA  110
      IF(FLON) 5,4,4                                                    PNA  120
    4 ISO=IB                                                            PNA  130
      GO TO 6                                                           PNA  140
    5 ISO=NM                                                            PNA  150
    6 CONTINUE                                                          PNA  160
      ITEMP=IFIX(TIME*600.+ROUND)                                       PNA  170
      IDA=ITEMP/14400                                                   PNA  180
      ITEMP=ITEMP-14400*IDA                                             PNA  190
      IH=ITEMP/600                                                      PNA  200
      IM=ITEMP-600*IH                                                   PNA  210
      ITEMP=IFIX(6000.*ABS(FLAT)+ROUND)                                 PNA  220
      ILA=ITEMP/6000                                                    PNA  230
      IAM=ITEMP-ILA*6000                                                PNA  240
      ITEMP=IFIX(6000.*ABS(FLON)+ROUND)                                 PNA  250
      ILO=ITEMP/6000                                                    PNA  260
      IOM=ITEMP-ILO*6000                                                PNA  270
      ICS=IFIX(10.*CSE+ROUND)                                           PNA  280
      ISP=IFIX(10.*SPD+ROUND)                                           PNA  290
      IEC=IFIX(ETV*100.+ROUND)                                          PNA  300
      WRITE (LOUT,71) IDA,IH,IM,ISA,ILA,IAM,ISO,ILO,IOM,ICS,ISP,IT,IEC  PNA  310
   71 FORMAT (1X,3I3,2X,A1,I2,1X,I4,1X,A1,I3,3(1X,I4),1X,A2,,1X,I6)     PNA  320
      RETURN                                                            PNA  330
      END                                                               PNA  340
```

```
      DIMENSION ALAT(10),ALONG(10)
      READ(5,102) NIN,NOUT,NSCRAT,NPLOT,IDELT,NLAT,NLONG,A
  102 FORMAT(7I5,F10.5)
      READ(5,101) (ALAT(I),I=1,NLAT)
      READ(5,101) (ALONG(I),I=1,NLONG)
  101 FORMAT(8F10.3)
      REWIND NIN
      REWIND NSCRAT
      CALL DRAWL(ALAT,ALONG,NLONG,NLAT,A,IDELT,NIN,NOUT,NSCRAT,NPLOT)
      END
```

```
      SUBROUTINE DRAWL(ALAT,ALONG,NX,NY,A,IDELT,NIN,NOUT,NSCAT,NPLOT)
C
C ALAT= ONE DIMENSIONAL ARRAY WITH NY.GE.2.AND.LE.10 IN DECIMAL DEGREES
C ASCENDING SEQUENCE(WHOLE OR HALF DEGREES ONLY)
C ALONG= ONE DIMENSIONAL ARRAY WITH NX.GE.2.AND.LE.10 IN DECIMAL DEGREES
C ACCENDING SEQUENCE (WEST OF GREENWICH NEQATIVE)(WHOLE OR HALF DEREES O
C A= SCALE IN INCHES/DEGREE
C IDELT= TIME INCREMENT IN MINUTES AT WHICH TIME TICKS WILL BE MADE
C (CHECKS NIN FOR INTEGRAL MULTIPLES OF THIS UNIT)
C NIN= INPUT TAPE NUMBER IN BCD MGG FORMAT (ONE FILE)
C NOUT=PRINTED OUTPUT TAPE NUMBER
C NPLOT= PLOTTING TAPE MUMBER
C NSCAT= SCRATCH TAPE NUMBER
C
      DIMENSION ALAT(10),ALONG(10),XF(10),YF(10),IBUF(1500),RFX(10)
     1 ,RFY(10),LMA(10),LNA(10)
      DIMENSION IA(4),IS(4)
      INTEGER WORD,WORD1,WORD2
      DATA PI,PW,EPS/3.1415926,28.0,1.0E-5/
      DATA WORD1,WORD2/2HDR,2HUP/
      DATA IA(1),IA(2),IA(3),IA(4),NWORD/2HSA,2HOM,2HLC,2HLA,4/
      DATA IS(1),IS(2),IS(3),IS(4),ISDFLT/2,12,0,5,11/
      RAD(X)=PI/180.0*X
C
C
      WRITE(NOUT,106) NIN,NOUT,NSCAT,NPLOT,A,IDELT,(ALAT(I),I=1,NY)
  106 FORMAT(74H1SUBROUTINE DRAWL HAS BEEN CALLED WITH THE FOLLOWING ARG
     1UMENT ASSIGNMENTS,,/5X,15HINPUT TAPE      ,I6,/5X,15HOUTPUT TAPE
     2 ,I6,/5X,15HSCRATCH TAPE    ,I6,/5X,14HPLOT TAPE      ,I7,//5X,17HTH
     3E PLOT SCALE IS,F10.4,28H  INCHES/DEGREE OF LONGITUDE,/5X,16HTIME 
     4TICKS EVERY,I11,10H   MINUTES,//5X,24HLATITUDE BOUNDS ARE      ,10F
     510.3)
      WRITE(NOUT,107) (ALONG(I),I=1,NX)
  107 FORMAT(5X,24HLONGITUDE BOUNDS ARE     ,10F10.3)
      DELT=FLOAT(IDELT)/24.0/60.0
      IF (DELT.LT.(1.0/24.0/60.0)) DELT=(10.0/24.0/60.0)
      DO 1 I=1,NX
      RFX(I)=RAD(ALONG(I))
      IF (RFX(I).LT.0.0) RFX(I)=2.0*PI+RFX(I)
    1 CALL MERK(0.0,RRR,RFX(I),XF(I),A,+1)
      DO 2 J=1,NY
      RFY(J)=RAD(ALAT(J))
    2 CALL MERK(RFY(J),YF(J),0.0,RRR,A,+1)
      IRX=NX-1
      IRY=NY-1
      DO 80 I=2,NY
      DIFF=YF(I)-YF(I-1)
      IF (DIFF.LE.PW) GO TO 80
      WRITE(NOUT,103)
  103 FORMAT(105H1YOUR CHOSEN LATITUDE MAP BOUNDS WOULD REQUIRE A PLOTTI
     1G SHEET LARGER THAN THIRTY INCHES.  PROGRAM STOPS.)
      STOP
   80 CONTINUE
C
   18 IZXA=0
      IZYA=0
   15 READ(NIN,161) ITEM,DAY,T1,T2,YLAT,YLATM,XLONG,XLONGM,ALPHA,V,WORD
      IF (DAY.GT.990.0) GO TO 16
  161 FORMAT(I1,2F3.0,F3.1,2X,F3.0,1X,F4.2,1X,F4.0,1X,F4.2,1X,F4.1,2X
     1 F3.1,1X,A2)
```

```
      TIME=DAY+T1/24.0+T2/24.0/60.0                                     DRW  610
      YLAT=RAD(YLAT+SIGN(YLATM,YLAT)/60.0)                              DRW  620
      XLONG=RAD(XLONG+SIGN(XLONGM,XLONG)/60.0)                          DRW  630
      IF (XLONG.LT.0.0) XLONG=2.0*PI+XLONG                              DRW  640
      CALL MERK(YLAT,Y,XLONG,X,A,+1)                                    DRW  650
      DO 11 J=1,IRX                                                     DRW  660
      DO 11 K=1,IRY                                                     DRW  670
      JJ=J                                                              DRW  680
      KK=K                                                              DRW  690
      IF (((X.LE.XF(J+1)).AND.(X.GT.XF(J))).AND.((Y.LE.YF(K+1)).AND.    DRW  700
     1(Y.GT.YF(K)))) GO TO 12                                           DRW  710
   11 CONTINUE                                                          DRW  720
      GO TO 18                                                          DRW  730
   12 IZX=JJ                                                            DRW  740
      IZY=KK                                                            DRW  750
      IF (WORD.EQ.WORD2) GO TO 18                                       DRW  760
      IF ((IZX.EQ.IZXA).AND.(IZY.EQ.IZYA)) GO TO 6                      DRW  770
      IC=4                                                              DRW  780
      WRITE(NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC               DRW  790
C     WRITE(NOUT,111) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC            DRW  800
  111 FORMAT(1X,4E12.5,2I6,3E12.5,I6)                                   DRW  810
    6 IF (WORD.NE.WORD1) GO TO 7                                        DRW  820
      IC=1                                                              DRW  830
      ITIME=IFIX(TIME/DELT+.49)                                         DRW  840
      TEST=TIME-FLOAT(ITIME)*DELT                                       DRW  850
      IF  (ABS(TEST).LT..0007) IC=3                                     DRW  860
      CO TO 77                                                          DRW  870
    7 IC=ISDFLT+20                                                      DRW  880
      DO 71 M=1,NWORD                                                   DRW  890
      IF (WORD.EQ.IA(M)) IC=IS(M)+20                                    DRW  900
   71 CONTINUE                                                          DRW  910
   77 WRITE(NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC               DRW  920
C     WRITE(NOUT,111) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC            DRW  930
      IZXA=IZX                                                          DRW  940
      IZYA=IZY                                                          DRW  950
      GO TO 15                                                          DRW  960
   16 IC=9                                                              DRW  970
      WRITE(NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC               DRW  980
      REWIND NSCAT                                                      DRW  990
      WRITE(NOUT,133)                                                   DRW 1000
  133 FORMAT(//15H FILE COMPLETE.)                                      DRW 1010
C                                                                       DRW 1020
C                                                                       DRW 1030
      READ (NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC               DRW 1040
      IF (IC.NE.9) GO TO 38                                             DRW 1050
      WRITE(NOUT,181)                                                   DRW 1060
  181 FORMAT (30HOALL POINTS OUTSIDE PLOT AREA.)                        DRW 1070
      RETURN                                                            DRW 1080
   38 JZ=1                                                              DRW 1090
      LMA(1)=1                                                          DRW 1100
      LNA(1)=1                                                          DRW 1110
      IF ((IRX.EQ.1).AND.(IRY.EQ.1)) GO TO 35                           DRW 1120
      LMA(1)=IZX                                                        DRW 1130
      LNA(1)=IZY                                                        DRW 1140
   34 READ(NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC                DRW 1150
      IF (IC.EQ.9) GO TO 35                                             DRW 1160
      DO 36 JI=1,JZ                                                     DRW 1170
      IF ((IZX.EQ.LMA(JI)).AND.(IZY.EQ.LNA(JI))) GO TO 34               DRW 1180
   36 CONTINUE                                                          DRW 1190
      JZ=JZ+1                                                           DRW 1200
```

```
      LMA(JZ)=IZX                                                       DRW 1210
      LNA(JZ)=IZY                                                       DRW 1220
      GO TO 34                                                          DRW 1230
   35 CONTINUE                                                          DRW 1240
C                                                                       DRW 1250
C         PLOTTING PACKAGE                                              DRW 1260
C                                                                       DRW 1270
  150 FORMAT(35HOCREATION OF A PLOT TAPE HAS BEGUN.)                    DRW 1280
      WRITE(NOUT,150)                                                   DRW 1290
      CALL PLOTS(IBUF,1500,NPLOT)                                       DRW 1300
      CALL PLOT(0.0,-30.0,-3)                                           DRW 1310
      CALL SYMBOL(1.0,9.9,.28,28HMACHINE ASSISTED SMOOTH PLOT,90.0,28)  DRW 1320
      CALL SYMBOL(1.42,12.1,.21,16HNOAA/AOML/MIAMI ,90.0,16)            DRW 1330
      XREF=0.0                                                          DRW 1340
      YREFLD=0.0                                                        DRW 1350
      WRITE(NOUT,110) JZ                                                DRW 1360
  110 FORMAT(33HONUMBER OF MAPS BEING PRODUCED IS,I6)                   DRW 1370
      DO 51 I=1,JZ                                                      DRW 1380
      REWIND NSCAT                                                      DRW 1390
      LM=LMA(I)                                                         DRW 1400
      LN=LNA(I)                                                         DRW 1410
      YREF=(PW+1.37-(YF(LN+1)-YF(LN)))/2.0-YREFLD                       DRW 1420
      YREFLD=(PW+1.37-(YF(LN+1)-YF(LN)))/2.0                            DRW 1430
      XREF=XREF+5.0                                                     DRW 1440
      CALL PLOT(XREF,YREF,-3)                                           DRW 1450
      CALL SYMBOL(-2.5,0.0,.14,4,0.0,-1)                                DRW 1460
      CALL PLOT(0.0,0.0,3)                                              DRW 1470
      JN=IFIX((RFY(LN+1)-RFY(LN))/(.5*PI/180.0)+1.0E-4)+1               DRW 1480
      IN=IFIX((RFX(LM+1)-RFX(LM))/(.5*PI/180.0)+1.0E-4)+1               DRW 1490
      XREF=FLOAT(IN-1)*A*.5+.25                                         DRW 1500
      YREF=YF(LN+1)-YF(LN)                                              DRW 1510
      YPAGE=0.0                                                         DRW 1520
      XPAGEM=0.0                                                        DRW 1530
      ITRIP=1                                                           DRW 1540
   53 DO 52 LP=1,IN                                                     DRW 1550
      XPAGE=FLOAT((LP-1)*(ITRIP))*A/2.0+XPAGEM                          DRW 1560
   52 CALL SYMBOL(XPAGE,YPAGE,.21,13,0.0,-2)                            DRW 1570
      XPAGEM=XPAGE                                                      DRW 1580
      DO 54 LP=1,JN                                                     DRW 1590
      PHI=RFY(LN)+.5*FLOAT(LP-1)*PI/180.0                               DRW 1600
      CALL MERK(PHI,Y,0.0,RRR,A,+1)                                     DRW 1610
      YPAGE=Y-YF(LN)                                                    DRW 1620
   54 CALL SYMBOL(XPAGEM,YPAGE,.21,13,90.0,-2)                          DRW 1630
      IF (ITRIP.EQ.(-1)) GO TO 55                                       DRW 1640
      ITRIP=-1                                                          DRW 1650
      GO TO 53                                                          DRW 1660
   55 CALL PASTA(2,RFX(LM),-.85,-.63,.21,0.0)                           DRW 1670
      CALL PASTA(2,RFX(LM+1),(XREF-1.1),-.63,.21,0.0)                   DRW 1680
      CALL PASTA (1,RFY(LN),XREF,-.11,.21,0.0)                          DRW 1690
      CALL PASTA(1,RFY(LN+1),XREF,(YREF-.11),.21,0.0)                   DRW 1700
   59 ITRIP=+1                                                          DRW 1710
   57 READ(NSCAT) X,Y,YLAT,XLONG,IZX,IZY,TIME,V,ALPHA,IC                DRW 1720
      IF (IC.EQ.9) GO TO 51                                             DRW 1730
      IF ((IZX.NE.LM).OR.(IZY.NE.LN)) GO TO 57                          DRW 1740
      X=X-XF(IZX)                                                       DRW 1750
      Y=Y-YF(IZY)                                                       DRW 1760
C     WRITE(NOUT,188) X,Y,IC                                            DRW 1770
  188 FORMAT(1X,2E14.5,I10)                                             DRW 1780
      ALPHA=AMOD((450.0-ALPHA),360.0)                                   DRW 1790
      TIME=(TIME+EPS)                                                   DRW 1800
```

```
      ITIME=IFIX(TIME)
      IQUIT=IC
      IF (IQUIT.GE.20)  IQUIT=2
      GO TO (61,62,63,64),IQUIT
   61 CONTINUE
      IF (ITRIP.EQ.-1) CALL PLOT(RXPAGE,RYPAGE,3)
      CALL PLOT(X,Y,2)
      GO TO 59
   62 IC=IC-20
      IANN=0
      IF (ITRIP.EQ.1) CALL WHERE(RXPAGE,RYPAGE,.5)
      CALL SYMBOL(X,Y,.07,IC,0.0,-1)
      XHR=((TIME-FLOAT(ITIME))*24.0+EPS)
      IHR=IFIX(XHR)
      XMIN=((TIME-FLOAT(ITIME)-FLOAT(IHR)/24.0)*60.0*24.0+EPS)
      XHR=FLOAT(IHR)*100.0+XMIN
   66 IF (ALPHA.GT.90.0.AND.ALPHA.LT.270.0) ALPHA=ALPHA+180.0
      ALPHA=AMOD(ALPHA,360.0)
      IF (ALPHA.GE.270.0) ALPHA=ALPHA-360.0
      ALPHA=ALPHA+SIGN(90.0,(-ALPHA))
      ALPHB=RAD(ALPHA)
      CALL NUMBER((X+.11*COS(ALPHB)),(Y+.11*SIN(ALPHB)),.07,XHR,ALPHA
     1,-1)
      IF (IANN.EQ.1) GO TO 67
      IF (ITRIP.EQ.1) CALL PLOT(RXPAGE,RYPAGE,3)
      ITRIP=-1
      GO TO 57
   63 CONTINUE
      IF (ITRIP.EQ.-1) CALL PLOT(RXPAGE,RYPAGE,3)
      CALL SYMBOL(X,Y,.07,13,ALPHA,-2)
      XHR =TIME-FLOAT(ITIME)
      IF (ABS(XHR).GT.2.0E-5) GO TO 59
      XHR=TIME
      CALL SYMBOL(X,Y,.07,11,ALPHA,-2)
      IANN=1
      GO TO 66
   67 CALL SYMBOL(999.0,999.0,.07,5H/0000,ALPHA,5)
      CALL PLOT(X,Y,3)
      GO TO 59
   64 CALL PLOT(X,Y,3)
      GO TO 59
   51 CALL SYMBOL(-2.5,0.0,.14,3,0.0,-1)
      XREF=XREF+5.0
      CALL SYMBOL(XREF,(YREF/2.0-1.40),.28,10HEND OF JOB,90.0,10)
      CALL PLOT(0.0,(YREF/2.0-1.40),-3)
      CALL PLOT(XREF,0.0,999)
      WRITE(NOUT,162)
  162 FORMAT(15H0PLOT FINISHED.)
      RETURN
      END
```

```
      SUBROUTINE PASTA(ITYPE,XX,X,Y,SIZE,ANGLE)
C
C THIS ROUTINE IS USED IN THE ANNOTATION OF A MERCATOR CHART.
C ITYPE=CONTROL CODE(ONE FOR LATITUDE AND TWO FOR LONGITUDE)
C XX=LATITUDE OR LONGITUDE IN RADIANS
C X=HORIZONTAL POSITION OF ANNOTATION
C Y=VERTICAL POSITION OF ANNOTATION
C SIZE=PLOT HEIGHT OF ANNOTATION
C ANGLE=ANGLE OF ANNOTATION
C
      DIMENSION XNOTE(4)
      DATA XNOTE(1),XNOTE(2),XNOTE(3),XNOTE(4)/1HN,1HE,1HS,1HW/
      DATA BLANC/1H /
      DATA PI/3.1415926/
      Z=XX
      IF (XX.GT.PI) Z=XX-2.0*PI
      IF (ITYPE.EQ.1) WORD=XNOTE(1)
      IF ((ITYPE.EQ.1).AND.(Z.LT.0.0)) WORD=XNOTE(3)
      IF (ITYPE.EQ.2) WORD=XNOTE(2)
      IF ((ITYPE.EQ.2).AND.(Z.LT.0.0)) WORD=XNOTE(4)
      Z=ABS(Z)
      Z=Z*180.0/3.1415926+1.0E-4
      NVAL=IFIX(Z)
      NMIN=IFIX((Z-FLOAT(NVAL))*60.0+1.0E-4)
      XNVAL=FLOAT(NVAL)
      XNMIN=FLOAT(NMIN)
      CALL NUMBER(X,Y,SIZE,XNVAL,ANGLE,-1)
      CALL SYMBOL(999.0,999.0,SIZE,BLANC,ANGLE,1)
      CALL NUMBER(999.0,999.0,SIZE,XNMIN,ANGLE,-1)
      CALL SYMBOL(999.0,999.0,SIZE,BLANC,ANGLE,1)
      CALL SYMBOL(999.0,999.0,SIZE,WORD,ANGLE,1)
      RETURN
      END
```

```
      SUBROUTINE MERK(PHI,Y,LAMDA,X,A,ITRIP)
C
C         ANGULAR VALUES IN RADIANS, A IN INCHES/DEGREE.
C         ITRIP.EQ.1 IF FORWARD TRANSFORM AND ITRIP.EQ.-1 IS INVERSE.
C         INTERNATIONAL SPHEROID
C         LAVELLE/AOML/JUNE 1972
C
      REAL LAMDA
      DATA WEE,PI/8.19918905E-2,3.1415926535/
      IF (ITRIP.EQ.-1) GO TO 1
      X=A*180.0/PI*LAMDA
      Y=A*180.0/PI*ALOG(TAN(PI/4.0+PHI/2.0)*((1.0-WEE*SIN(PHI))/(1.0+
     1 WEE*SIN(PHI)))**(WEE/2.0))
      RETURN
    1 XX=2.0*(ATAN(EXP(Y/A/180.0*PI))-PI/4.0)
      PHI=2.0*(ATAN(EXP(Y/A/180.0*PI)/((1.0-WEE*SIN(XX))/(1.0+WEE
     1 *SIN(XX)))**(WEE/2.0))-PI/4.0)
      PHI=2.0*(ATAN(EXP(Y/A/180.0*PI)/((1.0-WEE*SIN(PHI))/(1.0+WEE
     1 *SIN(PHI)))**(WEE/2.0))-PI/4.0)
      LAMDA=X/A/180.0*PI
      RETURN
      END
```

Data

```
   10    34     7     5     6     2    4
 100 0050                           094     5   TP
 100 0100    16 336    -59 091      094     5   SA
 100 0130                           097     5   TP
 100 0144    16 339    -59 052      097     5   SA
 100 0246    16 338    -58 598      097     5   SA
 100 0330    16 337    -58 560      097     5   SA
 100 0606    16 333    -58 423      097     5   SA
 100 0706    16 330    -58 371      097     5   SA
 100 0752    16 327    -58 332      097     5   SA
 100 0830                           002     5   TP
 100 0902    16 350    -58 297      002     5   SA
 100 1052    16 456    -58 299      002     5   SA
 100 1244    16 562    -58 296      002     5   SA
 100 1320                           270    63   TP
 100 1334    16 597    -58 309      270    63   SA
 100 1415                           004    55   TP
 100 1432    17 009    -58 351      004    55   SA
 100 1514    17 049    -58 353      004    55   SA
 100 1615                           269    63   TP
 100 1646    17 113    -58 380      269    63   SA
 100 1715                           090    05   TP
 100 1752    17 111    -58 397      090     5   SA
 100 1830    17 113    -58 362      090     5   SA
 100 1915                           270    05   TP
 100 1936    17 109    -58 337      270     5   SA
 100 2018    17 109    -58 385      270     5   SA
 100 2100                           268    05   TP
 100 2214    17 119    -58 509      268     5   SA
 100 2230                           266    05   TP
 101 0156    17 125    -59 134      266     5   SA
 101 0240                           266     5   NA
 101 0240    17 124    -59 178      266     5   SA
 101 0336    17 122    -59 238      266     5   SA
 101 0428    17 118    -59 292      266     5   SA
 101 0616    17 096    -59 397      266     5   SA
 101 0700                           266     5   NA
 101 0700    17 085    -59 451      266     5   SA
 101 0730                           273    05   TP
 101 0804    17 075    -59 515      273     5   SA
 101 0848    17 067    -59 560      273     5   SA
 101 0915                           277    05   TP
 101 1000    17 071    -60 039      277     5   SA
 101 1251                           282    05   TP
 101 1300                           292    05   TP
 101 1320                           270    05   TP
 101 1332    17 108    -60 257      270     5   SA
 101 1428    17 115    -60 311      270     5   SA
 101 1518    17 124    -60 351      270     5   SA
 101 1550
99999999
 101 1550                           038    16   TP
 101 1614    17 174    -60 351      038    16   SA
 101 1740    17 357    -60 215      038    16   SA
 101 1846    17 489    -60 113      038    16   SA
 101 1910
99999999
 101 1910                           090    06   TP
 101 1928    17 534    -60 054      090     6   SA
 101 1940                           180    05   TP
```

101 2124	17 441	-60 034	180	5	SA
101 2230			176	05	TP
101 2314	17 339	-60 040	176	5	SA
102 0000			180	05	TP
102 0056	17 241	-60 038	180	5	SA
102 0152	17 188	-60 037	180	5	SA
102 0240			180	5	NA
102 0240	17 142	-60 037	180	5	SA
102 0338	17 084	-60 041	180	5	SA
102 0530	16 575	-60 057	180	5	SA
102 0600			177	05	TP
102 0718	16 471	-60 078	177	5	SA
102 0756	16 432	-60 084	177	5	SA
102 0830			090	05	TP
102 0912	16 393	-60 070	090	5	SA
102 1104	16 393	-59 586	090	5	SA
102 1242	16 386	-59 520	090	5	SA
102 1338	16 381	-59 477	090	5	SA
102 1426	16 382	-59 438	090	5	SA
102 1440			083	05	TP
102 1524	16 386	-59 394	083	5	SA
102 1616	16 408	-59 339	083	5	SA
102 1654	16 403	-59 307	083	5	SA
102 1730			086	05	TP
102 1730			086	5	NA
102 1756	16 412	-59 247	086	5	SA
102 1842	16 427	-59 204	086	5	SA
102 1915			090	05	TP
102 1940	16 428	-59 141	090	5	SA
102 2030	16 428	-59 094	090	5	SA
102 2042	16 430	-59 081	090	5	SA
102 2215			094	05	TP
102 2224	16 437	-58 581	094	5	SA
103 0106	16 431	-58 426	094	5	SA
103 0115			311	16	TP
103 0250	16 585	-58 599	311	16	SA
103 0334	17 061	-59 085	311	16	SA
103 0438	17 172	-59 210	311	16	SA
103 0522	17 238	-59 303	311	16	SA
103 0545			180	05	TP
103 0628	17 234	-59 348	180	5	SA
103 0818	17 119	-59 355	180	5	SA
103 0830			177	05	TP
103 0854	17 069	-59 357	177	5	SA
103 1010	16 591	-59 363	177	5	SA
103 1130			092	05	TP
103 1256	16 512	-59 284	092	5	SA
103 1328	16 507	-59 251	092	5	SA
103 1436	16 507	-59 186	092	5	SA
103 1516	16 506	-59 145	092	5	SA
103 1626	16 507	-59 079	092	5	SA
103 1645			179	06	TP
103 1725			000	65	TP
103 1752	16 490	-59 065	000	65	SA
103 1815			179	05	TP
103 1848	16 510	-59 060	179	5	SA
103 1940	16 455	-59 060	179	5	SA
103 2040	16 394	-59 067	179	5	SA
103 2134	16 333	-59 069	179	5	SA
103 2145					

```
99999999
99999999
  XQT,V DRAW
    7       6    34      8     30      2      2    5.500
  16.500    18.00
 -61.00    -58.00
```

☆GPO 1974— 677-233/1212 REGION NO. 8