A System for the Precise Determination of Tow Distance and Tow Path in Offshore Resource Surveys

G. S. Jamieson

Fisheries Research Branch
Department of Fisheries and Oceans
Halifax, Nova Scotia B3J 2S7

March 1982

Canadian Technical Report of Fisheries and Aquatic Sciences No. 1035

Government of Canada Fisheries and Oceans
Gouvernement du Canada Pêches et Océans

Canadian Technical Report of Fisheries and Aquatic Sciences

These reports contain scientific and technical information that represents an important contribution to existing knowledge but which for some reason may not be appropriate for primary scientific (i.e. *Journal*) publication. Technical Reports are directed primarily towards a worldwide audience and have an international distribution. No restriction is placed on subject matter and the series reflects the broad interests and policies of the Department of Fisheries and Oceans, namely, fisheries management, technology and development, ocean sciences, and aquatic environments relevant to Canada.

Technical Reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report will be abstracted in *Aquatic Sciences and Fisheries Abstracts* and will be indexed annually in the Department's index to scientific and technical publications.

Numbers 1-456 in this series were issued as Technical Reports of the Fisheries Research Board of Canada. Numbers 457-714 were issued as Department of the Environment, Fisheries and Marine Service, Research and Development Directorate Technical Reports. Numbers 715-924 were issued as Department of Fisheries and the Environment, Fisheries and Marine Service Technical Reports. The current series name was changed with report number 925.

Details on the availability of Technical Reports in hard copy may be obtained from the issuing establishment indicated on the front cover.

Rapport technique canadien des sciences halieutiques et aquatiques

Ces rapports contiennent des renseignements scientifiques et techniques qui constituent une contribution importante aux connaissances actuelles mais qui, pour une raison ou pour une autre, ne semblent pas appropriés pour la publication dans un journal scientifique. Il n'y a aucune restriction quant au sujet, de fait, la série reflète la vaste gamme des intérêts et des politiques du Ministère des Pêches et des Océans, notamment gestion des pêches, techniques et développement, sciences océaniques et environnements aquatiques, au Canada.

Les Rapports techniques peuvent être considérés comme des publications complètes. Le titre exact paraîtra au haut du résumé de chaque rapport, qui sera publié dans la revue *Aquatic Sciences and Fisheries Abstracts* et qui figurera dans l'index annuel des publications scientifiques et techniques du Ministère.

Les numéros 1-456 de cette série ont été publiés à titre de Rapports techniques de l'Office des recherches sur les pêcheries du Canada. Les numéros 457-714, à titre de Rapports techniques de la Direction générale de la recherche et du développement, Service des pêches et de la mer, ministère de l'Environnement. Les numéros 715-924 ont été publiés à titre de Rapports techniques du Service des pêches et de la mer, Ministère des Pêches et de l'Environnement. Le nom de la série a été modifié à partir du numéro 925.

La page couverture porte le nom de l'établissement auteur où l'on peut se procurer les rapports sous couverture cartonnée.

Canadian Technical Report of

Fisheries and Aquatic Sciences 1035

March 1982

A SYSTEM FOR THE PRECISE DETERMINATION OF TOW DISTANCE AND TOW PATH IN OFFSHORE RESOURCE SURVEYS

by

G. S. Jamieson[1]

Fisheries Research Branch

Department of Fisheries and Oceans

Halifax, Nova Scotia B3J 2S7

[1]Present Address: Fisheries Research Branch

Department of Fisheries and Oceans

Pacific Biological Station

Nanaimo, British Columbia V9R 5K6

Cat. No. Fs 97-6/1035 ISSN 0706-6457

Correct citation for this publication:

Jamieson, G.S. 1982. A system for the precise determination of tow distance and tow path in offshore resource surveys. Can. Tech. Rep. Fish. Aquat. Sci. 1035: iv + 34 p.

CONTENTS

ABSTRACT

Jamieson, G.S. 1982. A system for the precise determination of tow distance and tow path in offshore resource surveys. Can. Tech. Rep. Fish. Aquat. Sci. 1035: iv + 34 p.

A system is described which allows accurate measurement and recording of Loran C navigational bearings. This allows *a posteriori* determination of geographic sample location, vessel speed over the sea bottom, and the length of the swath swept by the gear. The system utilizes a portable minicomputer with tape drive for data recording, and a larger host computer for later data analysis. System hardware and software are both described.

Key words: Loran C, sampling, biological surveys, stock abundance.

RÉSUMÉ

Jamieson, G.S. 1982. A system for the precise determination of tow distance and tow path in offshore resource surveys. Can. Tech. Rep. Fish. Aquat. Sci. 1035: iv + 34 p.

On décrit un système qui permet de mesurer précisement et d'enregistrer les relevés de navigation Loran C. On peut ainsi déterminer *a posteriori* la position géographique de l'échantillon, la vitesse du navire par rapport au fond, et l'étendue de la région couverte par l'engin échantillonneur. Le système utilise un mini-ordinateur à cassettes portatif pour l'enregistrement des données et un ordinateur de plus grande capacité pour l'analyse des donnés. Les composantes "hardware" et "software" du système sont aussi décrites.

INTRODUCTION

The quality of resource survey data for use in the quantitative estimation of a population's biomass is largely dependent on three variables:

1) the pattern of distribution of sample stations, relative to the resource;

2) the number of stations sampled; and

3) the area sampled at each station and its accurate measurement.

Accurate biomass estimation of deep-water, benthic populations has been particularly difficult when compared to sampling intertidal populations, as standard shallow-water survey procedures such as sample quadrats or line transects are usually impractical. Traditional offshore sampling equipment utilizes trawls, drags, or dredges towed for a fixed time period.

In ongoing studies to improve the quality of scallop resource surveys, an approach has been previously described (Jamieson and Chandler, 1980) which considers the spatial distribution of the resource. The number of stations sampled is generally determined by level of funding and available vessel time, while gear size and type are functions of vessel size and manpower constraints. This report considers measurement of the area sampled and presents a system, utilizing Loran C, designed to allow accurate measurement and recording of geographic sample location, drag speed over the sea bottom, and the length of the swath swept by the gear.

THE SYSTEM

Overview

Loran C signals are captured at discrete time intervals by a minicomputer on board the vessel and stored on magnetic tape for use in *a posteriori* analysis of vessel location. Real time vessel location, if desired, can be determined by use of a coordinate converter attached to an additional Loran C receiver. Data analysis on a mainframe, host computer involves initial editing of the data to ensure that tow beginning and tow end flags are present, followed by smoothing of the data to adjust for signal fluctuation, perhaps caused by wave-induced vessel rolling and speed alterations. The output from the Loran C receiver is averaged internally, but with the frequency of data recording utilized (every two seconds in "tow" mode; with an average towing speed of 3 kn, a distance of 3 m is travelled between recordings), small data value fluctuations prevent accurate tow distance estimation unless some additional data-averaging procedure is adopted. The bearings are then converted to latitude and longitude and the distance travelled determined between specified time periods. Distances are then summed over the duration of the tow. The shorter the time interval over which distance calculations are made, the greater the indicated tow distance, as more variation in vessel path is incorporated. The shortest tow distance is the straight-line track between tow end and tow beginning (longest time interval). Subjective data interpretation suggests that a time interval of 10 sec yields optimal results with towed gear, since the gear's path is less affected by slight fluctuations in the path of the towing vessel.

System components

The basic system consists of a Loran C receiver interfaced to a desk top, minicomputer, which records at specified time intervals the navigational bearings. Bearings are checked for quality, then stored with date and time in magnetic tape cartridges for subsequent data analysis. The system utilizes standard commercially available equipment but has as a unique feature, a specially designed interface unit (Model 8000, Loran C to computer interface) between the Loran C and minicomputer, developed under contract by Seimac Ltd., Bedford, Nova Scotia, Canada. This unit is now commercially available but is equipment-specific for compatability reasons and, as presently available, should be used with Internav Loran C receivers (Internav Ltd., Sydney, Nova Scotia) coupled to any RS232 or IEEE-488 terminal or computer. The computer used in the system described in this report is an HP9825 [Hewlett Packard (Canada) Ltd., Dartmouth, Nova Scotia].

The basic design layout is given in Figure 1, and a list of specific required hardware is given in Table 1.

Software

There are three basic procedures involved:

1) collecting and saving the data on tape;

2) dumping the data from the HP9825 to a mainframe, host computer - in this example, a CDC 171 (the HP9825 has insufficient memory to handle the analytical programs utilized);

3) analyzing the saved data to determine:

i) geographic location of the tow path,
ii) length of the tow path,
iii) tow duration (minutes),
iv) average vessel speed.

The first two procedures are programmed in HPL on the HP9825, whereas the last procedure is programmed in Fortran IV or 77 on the Cyber 171.

OPERATING PROCEDURES

Data collection

The hardware is set up as in Figure 1; and to initiate data collection, the procedure described in Appendix A is initiated.

Data transfer from tape to disk

The procedures described in Appendix B are for low-speed, dial-up communication between the minicomputer and host computer using an acoustic coupler. Two procedures are utilized, the first to establish a connection between the two computers and the second to transfer data from the tape cartridges to disc on the host computer.

Data analysis

Programs used in data analysis are given in Appendix C, and a trial run of test data is given in Appendix D. The output lists for each individual tow: 1) number of signal records; 2) tow duration (min); 3) calculated tow distances (m), using time intervals between navigational bearing recordings of 2 sec, 10 sec, and the entire tow duration; 4) the percent deviations of both the two-second based and ten-second based distances from the entire-tow based distance (these latter measures of distance deviation provide an indication of the straightness of the tow path without having to graphically plot the data); and 5) average gear speed (m/sec) over the sea floor.

SUMMARY

The system and procedures described in this report require no particularly sophisticated hardware or computer programming. The hardware required is readily available through commercial sources; and although the analysis software is designed for a CDC, it should be readily adaptable to most host computers. It improves the quality of surveys by increasing sample accuracy, as sampling error is magnified in the areal expansion of survey data to the area encompassed by the population under study.

ACKNOWLEDGEMENTS

I wish to thank Terry Deveau, Dirk Kelder and Dr. Robert Mohn, who together did most of the computer programming; Hugh MacPherson, President of Seimac Ltd., for his assistance in developing new hardware; Dr. Mike Eaton, Steve Grant, Bob O'Boyle and Wint Sparling for their assistance and support in system design and funding; and Gordon Kerr and Mark Lundy who helped make data analysis more efficient. Gary Collins and Drs. Ginette Robert and Frank Bernard kindly reviewed the manuscript.

REFERENCES

Jamieson, G.S. and R. Chandler. 1980. The potential for research and fishery performance data isopleths in population assessment of offshore, sedentary, contagiously distributed species. CAFSAC Res. Doc. 80/77, 32 p.

Table 1. Hardware descriptions of the components used in the capturing of Loran C signals and their subsequent transfer to a host computer.

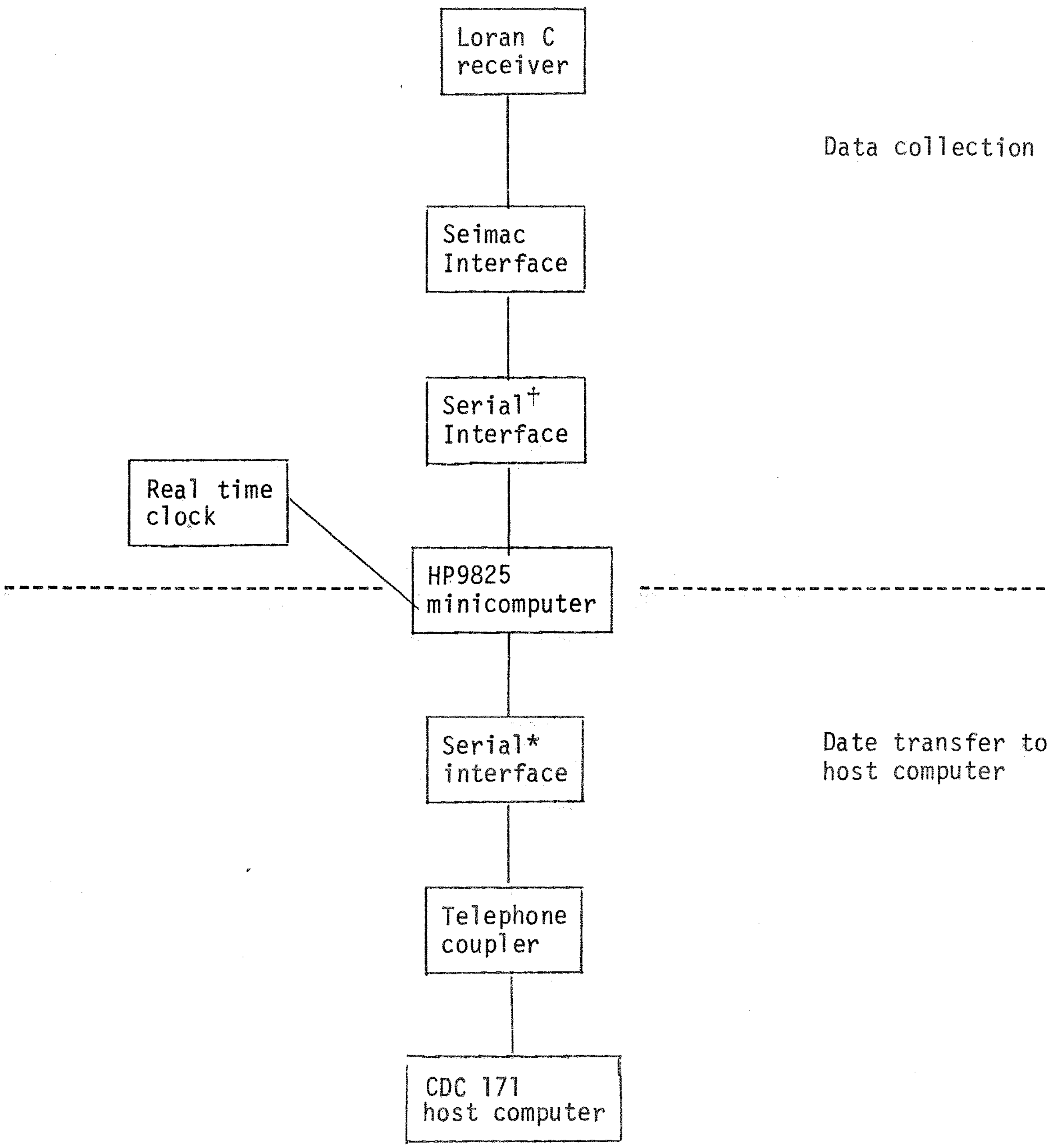
Loran C Equipment

1) Internav Loran LC123 receiver
2) Adjustable notch filters for 1)
3) 2.5 m whip antennae

Minicomputer Components

4) Hewlett Packard Model 9825S - desk top computer
5) Hewlett Packard Real Time Clock, Model 98035A, European time format option 002
6) Hewlett Packard Serial Interface, Model 98036A, female connector option 001
7) Tape cartridges for 4), No. 98200.
8) Thermal printer paper for 4), No. 9270-0479
9) Seimac Model 8000, Loran C to computer, interface
10) Telephone coupler

Figure 1. Schematic diagram of the hardware for data collection by the minicomputer and data transfer to the mainframe host computer.



†baud rate: 2400 (setting = 3)
*baud rate: 300 (setting = 7)

APPENDIX A

DATA COLLECTION

The programs in Table A-1 should be mounted in Files 0-2 on a specifically assigned tape cartridge, here termed "PROGRAM TAPE," so that it will be read as soon as the minicomputer is turned on. The operating procedure* is as follows:

1) Mount "PROGRAM TAPE" (Table A-1) in data cartridge slot (label is readable when cartridge is correctly mounted).

2) Turn on power to Loran C, then the HP9825, and finally the SEIMAC interface. [Note: on the serial interface (HP98036A), the select code "switch" and the "baud rate" switch should be set at 11 and 3 (2400 bps) respectively.]

3) You are then requested to mount a data tape to write on (preferably blank). Remove the program tape and replace it with the current data tape. Press the 'CONTINUE' key when the data tape is properly mounted.

4) If the data tape is blank (no data recorded), you will be asked to answer a few questions. Here are some typical responses:

What is the CRUISE ID?	P301	- then press 'CONTINUE'
What is the VESSEL ID?	PRINCE	- " " "
What year is this?	1980	- " " "
What is the data tape ID?	SCALLOPS	- " " "
What is the data tow, or tape, sequence number?	100	- " " "

5) When the program is "cold started" as above, it is in Cruise mode. To enter Tow mode, press Function Key 'f_1.' This key is labeled "Start Tow." (In Cruise mode, a position is logged once every 30 sec, while in Tow mode a position is logged once every 2 sec.)

6) You are requested to enter the Station Number (press 'CONTINUE' after typing the number). A check is made for sequential stations. If you must skip stations, just re-enter the new station number when requested.

---

*The comments in single quotes are labelled keys on the keyboard of the HP9825.

7) The program is now in Tow mode. When the tow is complete, press Function Key 'f_2,' labeled "Stop Tow," to return to Cruise mode.

8) Whenever it is necessary to interrupt the logging of data (either in Cruise mode or in Tow mode), such as at the end of the day or if the drag is caught on the bottom, simply press Function Key 'f_0.' This key is labeled "Halt" and stops the recording of data.

9) It is necessary to halt the program before interrupting the power to the HP9825. If the power is interrupted without first bringing the program to a halt, data will be lost.

10) After a halt, the program may be restarted where it left off by one of two procedures:

   a) if the power was not interrupted, simply press 'CONTINUE';

   b) otherwise, with the HP9825 and SEIMAC Interface off, begin with the "Cold Start" procedure from Step 1. After Step 3, the program will automatically restart (in the same mode).

11) Most of the time the program will display the current time in GMT [Grenwich Mean Time, or UCT (Universal Coordinated Time), as it is now called] and the estimated time until the next required data tape change (based on the current mode). When there are less than 90 sec to a tape change, the HP9825 will alert the operator with "beeps" at one-second intervals.

12) When the HP9825 is ready to accept the next sequential data cartridge, a mount message will be displayed and an alarm of four beeps per second will sound. With the correct tape mounted the operator can continue by pressing the 'CONTINUE' key.

13) Tapes are identified by an alphabetic label followed by a sequence number (e.g. SCALLOPS 1). When a blank tape is entered into the calculator, the operator should write this identification on the paper label of the cartridge.

14) The program requires a blank tape as the next sequential tape. You can force the program to accept a tape already containing data by pressing 'ERASE' instead of 'CONTINUE.' (Use caution here so not to overwrite a tape containing important data!)

15) The "Real Time Clock" will maintain its correct phase even while the HP9825 power is turned off (there are NiCad batteries in the clock itself). If, however, it becomes necessary to reset the clock to the correct GMT, this may be done by pressing Function Key 'f_5,' labeled "Set Time."

16) If this key (f_5) is pressed accidentally, exit the set time routine by pressing 'CONTINUE.'

17) Enter the time to set in advance. Respond to each question (i.e. Hour? Minute? etc.) with two digits, then press 'CONTINUE.'

18) The calculator will then wait until the time mark. At the time mark, press 'CONTINUE' to reset the clock to the time you previously entered. The program will then continue in whatever mode it was in before you pressed 'f_5.'
NOTE: When setting Month and Day of Month, remember that you are setting GMT and the date may not be exactly the same as your local month and day.

19) If an error occurs, consult the error booklet under the cover of the printer tape magazine on the HP9825. Correct the error, then press 'RUN' to restart the program. Some tape operations which result in an error may be retried by pressing 'CONTINUE'; these will be indicated in the display.

20) It is important to clean the tape drive head and drive wheel after every eight hours of use. Otherwise, buildup of dirt will cause tape errors. Computer instructions will describe how to do this.

The specified time period between data recordings can be modified. To change the time intervals in the tow and cruise modes, modify the value assigned to "I" in lines 41 and 42 (Table A-1; File 2) respectively. The more frequent the recording of data, the quicker tape capacity will be reached. Standard tapes will record for about 7 hrs in Tow mode and 108 hrs in Cruise mode. [The number of recording hours per tape cartridge can be calculated by multiplying 13,000 (i.e. 500 bytes/file times 13 files times 20 tracks) by the number of seconds between recordings, and then dividing by 3,600].

Table A-1. Programs used to transfer Loran C navigational bearings from Internav 123 receiver to HP9825 tape.

File #0

```
0: ldk 1
1: ldf 2
*31208
```

File #1

```
f0: *1→B

f1: *2→B

f2: *3→B

f5: *4→B
```

File #2

```
0: fxd 0
1: time 1000
2: "Tape Error
 Flag":0→O
3: on err "Error
  Recovery"
4: "Entries Per
 File":500→T
5: dim T$[3],
 V$[1],A$[19],
 A[0:T,1:3],I$[4
 ,10],Y,K,W,P,F,
 F[0:1]
6: dim B$[14],
 J$[4,10],Z,L,R,
 Q,G,G[0:1]
7: "Validation
 Character":"1"→
 V$
8: "Current Tow
 Number":0→W
9: "Previous
 Tow Number":0→P
10: "Files Per
 Track":13→F
11: on key "Moun
 ted";-4→J
12: dsp "Mount
 the current
 data tape"
13: if J=1;gto +
 4
14: wait 2000;
 dsp "Press CONT
 INUE when ready
 "
15: if J=1;gto +
 2
16: wait 2000;
 jmp J
17: gto "A Tape"
18: "Mounted":ke
 y→R
19: if R=29 or
 R=157;0→R;gto +
 2
20: if R#25 and
 R#153;kret
21: on key ;1→J;
 kret
22: "A Tape":gsb
  "Read Header"
23: jmp J
24: gto "Good
 Header"
25: trk 1;rew;
 mrk 1,126;0→K;
 1→F[0];0→F[1]
26: trk 0;rew;
 mrk 1,126;mrk
 1,24(T+1)
27: ent "What
 is the Cruise
 ID",I$[1]
28: ent "What
 is the ID of
 this Vessel",
 I$[2]
29: ent "Enter
 the current
 year(s)",I$[3]
30: ent "What
 is the Data
 Tape ID",I$[4]
31: ent "Whct
 is the Data
 Tape Sequence
 #",Y
32: gsb "Write
 Header"
33: trk 0;gto
 "Set Interval"
```

.../Contd.

Table A-1 Contd...

```
34: "Good Header
 ":trk 0;ldf 0,
 I$,Y,K,W,P,F,
 F[*]
35: if K<0;gto +
 2
36: trk K;ldf
 F[K],A[*];A[0,
 1]→N;gto "Set
 Interval"
37: dsp "This
 tape is full";
 beep
38: wait 5000;
 beep
39: gsb "End Of
 Tape"
40: trk K
41: "Set Interva
 l":2→I;if not
 W=0;gto +2
42: 30→I
43: wrt 9,"U1H,
 U1=01,U1P1000/
 U1G"
44: oni 9,"Time
 Mark"
45: eir 9;0→L
46: "Idle Loop":
 0→B
47: jmp B
48: gto "Pause"
49: gto "Start
 Tow"
50: gto "Stop
 Tow"
51: gto "Set
 Time"
52: gto "Get
 Location"
53: "Start Tow":
 if W=0;gto +2
54: dsp "Current
 ly in tow"&str(
 W);gto "Idle
 Loop"
55: eir 9,0;ent
 "Enter the Stat
 ion Number",W
56: if P=0 or P+
 1=W;gto "Seq
 Station"
57: "Out Of Seq"
 :beep
58: prt "Last
 Station was",P
59: ent "This
 Station Number
 ?",R
60: if R=0;gto
 "Out Of Seq"
61: if R#P+1
 and R#W;R→W;
 gto "Out Of
 Seq"
62: R→W
63: "Seq Station
 ":A[N,3]→A[N+
 1→N,2];0→A[N,
 3];W→A[N,1]
64: "Record Chan
 ge":if N<T;gto
 +4
65: gsb "Write
 File"
66: 0→N;trk K
67: gto "Set
 Interval"
68: trk K
69: gto "Set
 Interval"
70: "Stop Tow":i
 f not W=0;gto +
 2
71: dsp "Not
 currently in
 tow";gto "Idle
 Loop"
72: eir 9,0;A[N,
 3]→A[N+1→N,2];
 1→A[N,3];W→A[N,
 1]
73: beep;dsp
 "End of Tow";
 wait 500;beep
74: W→P;0→W;gto
 "Record Change"
75: "Pause":eir
 9,0;N→A[0,1];
 trk K
76: dsp "Crunch
 ... Crunch ..."
77: rcf F[K],A[*
 ]
78: gsb "Write
 Header"
79: dsp "To rest
 art press CONTI
 NUE"
80: stp
81: gto +2
82: gsb "Await
 Mount"
83: gsb "Read
 Header"
84: trk K;if
 J#1;gto -2
85: gto +6
86: trk K;ldf 0,
 J$,Z,L,R,Q,G,
 G[*]
```

.../Contd.

Table A-1 Contd...

```
87: for J=1 to 4
88: if I$[J]#J$[
 J];gto -6
89: next J
90: if Z#Y or
 L#K or R#W or
 Q#P or G#F or
 G[0]#F[0] or
 G[1]#F[1];gto -
 8
91: gto "Set
 Interval"
92: "Get Locatio
 n":eir 9,0
93: 0→L
94: N+1→N
95: S+100(M+100(
 H+100(Q+100D)))
 →A[N,3]
96: wtc 11,1;
 wtb 11,5;0→Q
97: wtc 11,0;
 wrt 11,"A";wait
  10
98: "Loran C":Q+
 1→Q;wtc 11,1;
 wtb 11,23
99: rdb(11)→C;
 if bit(1,C)=0;
 gto +0
100: wtb 11,5;
 wtc 11,0
101: char(rdb(11
 ))→A$[Q]
102: if Q<19;
 gto "Loran C"
103: if A$[8,
 8]=V$ and A$[17
 ,17]=V$;gto
 "Valid Data"
104: dsp "LORAN
 C not ready -
 wait"
105: N-1→N
106: gto "Set
 Interval"
107: "Valid Data
 ":val(A$[1,7])/
 100→A[N,1]
108: val(A$[10,
 16])/100→A[N,2]
109: if N<T;gto
 "Set Interval"
110: gsb "Write
 File"
111: 0→N;trk K
112: gto "Set
 Interval"
113: "Time Mark"
 :wrt 9,"T";rdb(
 9)→C
114: if not bit(
 0,C);gto "Texit
 "
115: wrt 9,"R";
 red 9,D,Q,H,M,S
116: if not (L+
 1→L)<I;5→B
117: T(F(2-K)-
 F[K]+1)-N→R;IR/
 60→Z
118: "min"→T$;
 if Z<60;gto +2
119: "hrs"→T$;Z/
 60→Z;gto +3
120: if not Z<1;
 gto +2
121: "sec"→T$;
 60Z+38→Z;beep
122: dsp "GMT"&s
 tr(H)&str(M)&st
 r(S)&" - Tape
 Ends:"&str(Z)&T
 $
123: "Texit":eir
  9;iret
124: "Write File
 ":N→A[0,1]
125: rcf F[K],
 A[*]
126: mrk 1,24(T+
 1),R
127: if -R=F[K];
 gto "End Of
 Track"
128: F[K]+1→F[K]
129: ret
130: "Write Head
 er":0→R
131: trk R;rcf
 0,I$,Y,K,W,P,F,
 F[*]
132: jmp 2(R+
 1→R)-3
133: ret
134: "End Of
 Track":if K>0;
 gto "End Of
 Tape"
135: F[0]→F;1→F[
 1]
136: 1→K;trk K;
 fdf 1
137: mrk 1,24(T+
 1);gto "Write
 Header"
138: "End Of
 Tape":-1→K
139: gsb "Write
 Header"
```

.../Contd.

Table A-1 Contd...

```
140: 0→K;Y+1→Y;
 0→F[1];1→F[0]
141: "Request
 Mount":gsb "Awa
 it Mount"
142: gto "Check
 Tape"
143: "Await Moun
 t":on key "New
 Tape";-10→J
144: dsp "Mount
 "&I$[4]&str(Y)
145: for L=1 to
 8
146: if J=1;ret
147: beep;wait
 250
148: next L
149: dsp "press
 CONTINUE when
 ready"
150: for L=1 to
 8
151: if J=1;ret
152: beep;wait
 250
153: next L
154: jmp J
155: ret
156: "New Tape":
 key→R
157: if R=157
 or R=29;0→R;
 gto +2
158: if R#25
 and R#153;kret
159: 1→J;on key
 ;kret
160: "Check Tape
 ":gsb "Read
 Header"
161: jmp J
162: dsp "Tape
 is NOT BLANK -
 try another";
 wait 5000;gto
 "Request Mount"
163: trk 1;rew;
 mrk 1,126;0→K;
 1→F[0];0→F[1]
164: trk 0;rew;
 mrk 1,126;mrk
 1,24(T+1)
165: gto "Write
 Header"
166: "Set Time":
 eir 9,0;wrt 9,
 "B"
167: dsp "Enter
 the time to
 set in advance"
 ;wait 2000
168: " "→B$[1,
 14]
169: ent "Day
 of month ? (DD)
 ",B$[1,2]
170: if flg13;
 gto "Set Interv
 al"
171: ent "Month
 ? (MM)",B$[4,5]
172: if flg13;
 gto "Set Interv
 al"
173: ent "Hour
 00 - 23 (HH)",
 B$[7,8]
174: if flg13;
 gto "Set Interv
 al"
175: ent "Minute
 ? (MM)",B$[10,
 11]
176: if flg13;
 gto "Set Interv
 al"
177: ent "Second
 ? (SS)",B$[13,
 14]
178: if flg13;
 gto "Set Interv
 al"
179: dsp "Press
 CONTINUE on
 time MARK"
180: stp
181: wrt 9,"S",
 B$
182: dsp "Please
 WAIT"
183: wait 30000;
 wrt 9,"B"
184: gto "Set
 Interval"
185: "Error Reco
 very":
186: on err "Err
 or Recovery"
187: if rom#0
 or ern#41;gto +
 3
188: dsp "TAPE
 is not securely
 in DRIVE"
189: beep;wait
 1000;jmp erl-
 cln
190: if rom#0
 or ern#42;gto +
 2
```

.../Contd.

Table A-1 Contd...

```
191: dsp "This
 tape is WRITE
 PROTECTED";beep
 ;wait 1000;jmp
 erl-cln
192: if rom#0;
 gto +7
193: if ern<43
 or ern>49;gto +
 6
194: prt "Unreco
 verable    TAPE
  ERROR"
195: dsp "retry=
 CONTINUE , next
  tape = RUN"
196: stp
197: jmp erl-cln
198: 1→O;gto 7
199: if rom=69
 and ern=4;prt
 "LORAN C is
 DOWN";beep;beep
 ;gto "Pause"
200: if rom=69
 and ern=8;gto
 47
201: if rom#0
 or ern#65;gto
 "Print Error"
202: "Handle
 Blank Tape":if
 erl#cln+4;gto
 "Print Error"
203: gto +10
204: "Read Heade
 r":dsp "Crunch
 ... Crunch ..."
205: if R=0;jmp
 5-O
206: 0→O;trk 0;
 rew;idf Q,H,M,S
207: 1→J;if Q=0
 and H=3 and
 S=126;ret
208: dsp "To
 ERASE tape pres
 s CONTINUE"
209: stp
210: gto +2
211: trk 0;rew;
 mrk 1,120;trk
 1;rew;mrk 1,
 120;0→O
212: trk 0;rew;
 ert 0;trk 1;
 rew;ert 0
213: 2→J;ret
214: "Print Erro
 r":
215: rom→R;if
 R=0;32→R
216: prt "Fatal
 Error "&char(R)
 &str(ern)&" in
 line "&str(erl)
217: gto "Pause"
218: end
*3234
```

APPENDIX B

The program listed in Table B-1 is used to establish communication between the minicomputer and host computer, while the program listed in Table B-2 transfers the data to a disc file. The following operating procedure* should be followed:

1) Define a file "Data" on the desired account in the host computer through a normal terminal.

2) Dial the host computer, get a signal, and mount the phone in the coupler. Place the 'Program tape' in the HP9825. [Note: On the serial interface (HP98036A), the 'select code' switch and the 'baud rate' switch should be set at 11 and 7 (300 bps) respectively.]

3) Type in 'ldF(fno)' (Table B-1), then press 'EXECUTE' and 'RUN.' [Note: (fno) = file number of this program on the tape cartridge.]

4) When the user number is asked for, enter it and the password, then press 'STORE' and 'RESET.'

5) Type in 'ldF(fnl)' (Table B-2), then press 'EXECUTE.'

6) Replace the 'Program tape' with the 'Data tape,' then press 'REWIND' and 'Run.'

7) The end of the data tape will be announced on the HP9825, and if there is another sequential data tape to be transmitted as part of this run, load the next tape and press 'REWIND,' then 'CONTINUE.'

8) After transmission is complete, log on to the host computer through a normal terminal again, and type in "CHANGE, pfn = DATA," where pfn is the permanent file name of the transmitted data.

9) If additional runs are to be transmitted, type in "CLEAR," "DEFINE, DATA," then "CLEAR" again; and then repeat 6).

---

*The comments in double quotes are part of the Cyber Control Language (CCL) used with the Network Operating System (NOS) of the Cyber 171 host, which was used in this instance.

Table B-1. Program used to establish communication between the minicomputer and the host computer.

```
0: dim A[0:500,
 1:3],I$[4,10],
 Y,Q,W,P,R,F[(
 ]
1: "New Tape":ds
 p "Please Wait"
2: fxd 0;wrt 11,
 "NEW,PRIM"
3: wait 5000;
 wrt 11,"TEXT"
4: wait 5000
5: -1→S
6: wrt 9,"R";
 red 9,D,Q,H,M,V
7: prt "Start
 :"&str(H-3)&"/
 "&str(M)&"/"&st
 r(V)
8: "New Track":S
 +1→S;if S>1;
 gto "Next Tape"
9: trk S;rew
10: ldf 0,I$,Y,
 Q,W,P,R,F[*]
11: prt "Tape :
 "&I$[4]&str(Y)
12: prt "Track
 :"&str(S)
13: for K=1 to
 15
14: fdf K
15: idf F,T
16: if T=0;gto
 "New Track"
17: ldf K,A[*]
18: prt "File
 :"&str(K)&",
 "&str(A[0,1])
19: for I=1 to
 A[0,1]
20: fxd 1;wrt
 11,str(A[I,1])&
 ","&str(A[I,
 2])&","&str(A[I
 ,3])
21: fxd 0;dsp I
22: next I
23: next K
24: gto "New
 Track"
25: "Next Tape":
26: wait 5000;
 wtb 11,char(20)
 ;wtb 11,char(13
 )
27: wait 5000
28: wrt 11,"ATTA
 CH,DATA/M=W"
29: wait 5000
30: wrt 11,"SKIP
 EI,DATA"
31: wait 5000
32: wrt 11,"COPY
 EI,PRIM,DATA"
33: wait 10000;
 wrt 11,"RETURN,
 DATA";wait 3000
34: wrt 11,"DAYF
 ILE,A";wrt 11,
 "REPLACE,A=DAYF
 ILE"
35: dsp "This
 tape is FINISHE
 D"
36: wrt 9,"R";
 red 9,A,Q,G,N,U
37: prt "Stop
 :"&str(G-3)&"/
 "&str(N)&"/"&st
 r(U)
38: if A#D;G+
 24→G
39: fxd 2;prt
 "Time :"&str(60
 (G-H)+N-M+(U-
 V)/60)
40: for I=1 to
 10
41: beep
42: wait 500
43: next I
44: stp
45: gto "New
 Tape"
46: end
*26259
```

Table B-2. Program used to transfer data from the minicomputer to a disc file on the host computer.

```
0: "Number of
 32 byte lines
 to save":50→L
1: dim A$[32],
 B$[L,32],C$[85]
2: 0→I;0→J;0→S;
 0→D
3: wsc 11,32+4+
 2+1
4: on key "Keybo
 ard"
5: "Login Proced
 ure - AUTO BAUD
 ":
6: dsp "Please
 Wait"
7: if bit(1,rss
 11)=1;gto +5
8: wait 3000
9: if bit(1,rss
 11)=1;gto +3
10: wrt 11,char(
 13)
11: gto -4
12: band(127,
 rdb(11))→A;if
 A=0;gto -2
13: if A=13;gto
 -6
14: if A=10;wait
  500;gto -7
15: dsp "OK -
 communication
 established"
16: gto "Modem"
17: "Keyboard":
18: key→K;band(1
 27,asc K)→C
19: if K#19 and
 K#147;gto +3
20: gsb "Print
 B$"
21: kret
22: if K=159;
 2→S;kret
23: if K=1 or
 K=129;1→S;kret
24: if K=158;
 26→C
25: if K=20 or
 K=148;8→C
26: if K=22 or
 K=150;96→C
27: if K=7 or
 K=135;126→C
28: if K=15 or
 K=143;9→C
29: if K=16 or
 K=144;10→C
30: if K=18 or
 K=146;20→C
31: if K=23 or
 K=151;127→C
32: wtb 11,char(
 C)
33: kret
34: "Print B$":
35: J→K
36: prt B$[JmodL
 +1→J];if J#K;
 gto +0
37: ret
38: "Modem":
39: if S=3 and
 D=1;gto +2
40: if S#0;gto
 "Stop"
41: if bit(1,
 rss 11)=1;gto +
 2
42: gto "Modem"
43: band(127,
 rdb(11))→A;if
 A=0 or A=13;
 gto +8
44: if A=10;gto
 +4;if D=1;2→D
45: char(A)→A$[I
 +1→I];if A=8;
 if I>1;" "→A$[(
 I-2→I)+1]
46: dsp A$
47: if I<32;gto
 +4
48: if D#0;A$&C$
 →C$;gto +2
49: JmodL+1→J;
 A$→B$[J]
50: 0→I;""→A$
51: gto "Modem"
52: "Stop":if
 S#1;gto +2
53: stp ;0→S;
 gto "Modem"
54: if S=2;0→I;
 ""→A$;avm→V;
 nal→N;3→S;1→D;
 wtb 11,char(13)
 ;gto +6
55: 1→D;if len(C
 $)<=2;gto +5
56: if C$[1,1]="
 *";gto "Out"
57: if num(C$[1]
 )=0;gto +3
58: dsp C$;on
 err "Err";store
  C$,nal
59: if avm<250;
 beep;dsp "Insuf
 ficient MEMORY"
 ;stp
```

.../Contd.

Table B-2 Contd...

```
60: ""→C$;gto
 "Modem"
61: "Out":V-avm→
 V
62: for F=0 to
 9999
63: fdf F;idf F,
 Y,C,Q;if Q;next
  F
64: mrk 1,V+500,
 Z;if Z<0;beep;
 prt "Not enough
  TAPE,",V,"byte
 s needed";stp
65: rcf F,1N;
 0→D;0→S;gto
 "Modem"
66: "Err":"%"&C$
 →C$
67: if not (pos(
 C$,";")→X);gto
 -9
68: "%"→C$[X,X];
 gto -1
69: end
*30052
```

APPENDIX C

The raw data should first be checked to ensure that the dump from the HP9825 was successful, and that station numbers were correctly inserted at both the beginning and end of tow. Experience has indicated that control flags may accidentally be inserted by the operator at the wrong time, making a check of the "saved" data with the vessel's tow record log often a worthwhile procedure. To obtain a less bulky listing of the raw data, submit the program in Table C-1. To obtain a sequential listing of the control flags and the data at the following adjacent time recording, submit the program in Table C-2.

After any necessary data editing has been completed, the data can be smoothed by utilizing the program in Table C-3. Sample output is shown in Table C-4. This procedure utilizes a weighted average to smooth the data with 19 triangular weights.

A smooth startup and ending is incorporated; this utilizes a smaller triangle at each end of the data string to avoid spurious oscillation.

The data are now ready for final analysis, utilizing the program in Table C-5. For each tow, output is as follows:

	Tow No.	No. Records	Tow Duration (D) (min)	TOW DISTANCE A	B	C	% Diff. $\frac{A-C}{C}$ *100	% Diff. $\frac{B-C}{C}$ *100	Avg. Speed (m/sec) (B/(D*60))
e.g.	1	234	9.2	1195.44	993.44	899.92	32.84	10.39	1.80

where A = total distance with 2 sec time intervals;
B = total distance with 10 sec time intervals;
C = total straight-line distance between tow beginning and end.

Output from eleven research tows is shown in Table C-6. By comparing the percent differences between the tow distance calculated from every fifth navigational recording and that calculated from the beginning/end of tow, it is evident that Tow 5 was considerably curved. This comparison allows rapid detection of those tows which were not in a straight line.

Table C-1. Program to produce a less-bulky listing of the raw data. Language: Fortran 77.

```
      PRØGRAM PLIST(INPUT.ØUTPUT,TAPE5,TAPE6)
C     THIS PRØGRAM IS USED TØ PRØDUCE A LESS BULKY
C     LISTING ØF THE RAW DATA BY WRITING THE DATA
C     IN HØRIZØNTAL RØWS ACRØSS THE PAGE RATHER
C     THAN AS ØNE VALUE PER LINE.
      DIMENSIØN X(49,4),Y(49,4)
      INTEGER Z1(49,4),Z2(49,4),DATETM,FLAG
      REAL LØRANA,LØRANC
      I=0
      N=49
      J=1
      FLAG=0
      CALL ZERØ(X.Y.Z1,Z2)
      NCØUNT=0
10    READ(5,*,END=12)LØRANA.LØRANC,DATE
      IDATE=IFIX(DATE)
      ENCØDE(10,15,DATETM)IDATE
15    FØRMAT(I10)
      NCØUNT=NCØUNT+1
      I=I+1
      IF(I.GT.49)THEN
         J=J+1
         I=1
      END IF
      IF(J.GT.4)THEN
         M=4
         GØTØ30
      END IF
20    X(I,J)=LØRANA
      Y(I,J)=LØRANC
      IF(DATETM.NE.0.0)THEN
         DECØDE(10,25,DATETM)Z1(I,J),Z2(I,J)
25       FØRMAT(I4,I6)
      ELSE
         Z1(I,J)=DATETM
      END IF
      GØTØ10
  12     FLAG=1
         N=I
         M=J
         GØTØ30
30    WRITE(6,40)
40    FØRMAT('1'/4(4X,'LØRAN',5X,'LØRAN',2X,'DATE',4X,'TIME'))
      WRITE(6,50)((X(IX,IY),Y(IX,IY),Z1(IX,IY),Z2(IX,IY),IX=1,
     $N),IY=1,M)
50    FØRMAT(49(1X,4(F8.1,2X,F8.1,2X,I4,2X,I6,1X)/))
      IF(FLAG.EQ.1)THEN
         WRITE(6,60)NCØUNT
60       FØRMAT(/,1X,' TØTAL LINES CØPIED=',I6)
         STØP
      END IF
```

Table C-1 Contd...

```
      CALL ZERØ(X,Y,Z1,Z2)
      I=1
      J=1
      GØTØ20
      END
      SUBRØUTINE ZERØ(X,Y,Z1,Z2)
      DIMENSIØN X(49,4),Y(49,4)
      INTEGER Z1(49,4),Z2(49,4)
      DØ 100 I=1,49
         DØ 75 J=1,4
            X(I,J)=0.0
            Y(I,J)=0.0
            Z1(I,J)=0
            Z2(I,J)=0
75       CØNTINUE
100   CØNTINUE
      RETURN
      END
```

Table C-2. Program used to obtain a sequential listing of the control flags. Language: Fortran IV.

```
      PROGRAM LIST (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE8)
C     * THIS PROGRAM LISTS SEQUENTIALLY THE BEGINNING AND END FLAGS
C     *  OF THE TOWS, PLUS THE FIRST LORAN RECORD AFTER EACH OF
C     *  THE ABOVE SO THAT THE TIME OF OCCURRENCE CAN BE CHECKED.
      Y = 0
   20 READ (7,*) X1,X2,X3
      IF (EOF(7).NE.0) GO TO 30
      IF (Y.EQ.1) WRITE (8,40) X1,X2,X3
      Y = 0
      IF (X1.LT.500) Y = 1
      IF (X1.LT.500) WRITE (8,40) X1,X2,X3
      GO TO 20
   30 STOP
   40 FORMAT (3(F15.1),1X))
      END
```

Table C-3. Program used to smooth the raw data with 19 triangular weights. Language: Fortran IV.

```
      PROGRAM SMDATA (OUTPUT,TAPE6,TAPE7,TAPE8,TAPE5,DEBUG=TAPE5)
C
C     THIS PROGRAMME UTILIZES A WEIGHTED AVERAGE TO SMOOTH THE
C     DATA.WEIGHTS ARE TRIANGULAR AND INVOLVE 19 POINTS. A SMOOTH
C     STARTUP AND ENDING IS INCORPORATED TO AVOID SPURIOUS
C     OSSCILATION AT EACH END OF THE DATA STRING.
C
      EQUIVALENCE (XIN,Y)
      DIMENSION XIN(2600), YIN(2600), X(2600), Y(2600), WGT(19), XL(2600
     $)
      DATA WGT/1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,9.,8.,7.,6.,5.,4.,3.,2.,1.
     $/
      NWGT = 19
C    *
C     INITIALIZE
C
   20 CONTINUE
      N = 0
      NN = 0
      ISW = 0
C
C     READ INPUT FILE
C
   30 READ (7,*) XX,YY,TIME
      IF (EOF(7).NE.0) GO TO 90
      NN = NN+1
      IF (XX.GT.500.) GO TO 50
C     OUTPUT CORRECT STATION NO.
      IF (XX.GT.10.) GO TO 40
      XX = XX*10.
      YY = AINT(YY+.5)
      YY = YY/10.
   40 CONTINUE
      IF (TIME.EQ.1.) GO TO 80
C
C     BEGINNING OF TOW
C
      WRITE (8,110) XX,YY,TIME
      ISW = 1
      GO TO 30
C
C     MANIPULATE AND STORE DATA
C
   50 CONTINUE
      IF (ISW.EQ.0) GO TO 30
      IF (XX.GT.10.) GO TO 70
   60 XX = AINT(XX+.5)
      YY = AINT(YY+.5)
      XX = XX/10.
      YY = YY/10.
   70 CONTINUE
```

.../Contd.

Table C-3 Contd...

```
      N = N+1
      XIN(N) = XX
      YIN(N) = YY
      XL(N) = TIME
      GO TO 30
C
C     END OF TOW
C
   80 CALL SMOOTH (XIN,X,N,WGT,NWGT)
      CALL SMOOTH (YIN,Y,N,WGT,NWGT)
      WRITE (8,100) (X(I),Y(I),XL(I),I=1,N)
      WRITE (8,110) XX,YY,TIME
      GO TO 20
   90 STOP
  100 FORMAT (2(F10.2,1H,,1X),F12.0,1H,)
  110 FORMAT (3(F8.1,1H,,1X))
      END
      SUBROUTINE SMOOTH (X,Y,N,V,NV)
      DIMENSION X(2600), Y(2600), V(19)
      NSHIFT = NV/2
      NSTART = NSHIFT+1
      NEND = N-NSHIFT
      VS = 0.
      DO 10 I=1,NV
         VS = VS+V(I)
   10 CONTINUE
      DO 20 I=NSTART,NEND
         Y(I) = 0
         DO 20 J=1,NV
            Y(I) = Y(I)+V(J)*X(I-NSHIFT+J-1)/VS
   20 CONTINUE
      DO 40 I=1,NSHIFT
         K = NSHIFT+I
         Y(I) = 0.
         VS = Y(I)
         DO 30 J=1,K
            Y(I) = Y(I)+X(J)*V(NV-K+J)
            VS = VS+V(NV+1-J)
   30    CONTINUE
         Y(I) = Y(I)/VS
   40 CONTINUE
      N1 = NEND+1
      K = NV
      DO 60 I=N1,N
         Y(I) = 0.
         VS = Y(I)
         K = K-1
         DO 50 J=1,K
            Y(I) = Y(I)+X(I+J-NSHIFT-1)*V(J)
            VS = VS+V(J)
   50    CONTINUE
         Y(I) = Y(I)/VS
   60 CONTINUE
      RETURN
      END
```

Table C-4. Sample data of smoothing program (Table C-3).
A = input data. B = output data.

A

```
1.0, 30843.5, 0.0
12273.7, 30945.7, 2907185909.0
12273.7, 30945.7, 2907185911.0
12273.7, 30945.7, 2907185914.0
12273.7, 30945.7, 2907185916.0
12273.7, 30945.7, 2907185918.0
12273.7, 30945.8, 2907185921.0
12273.6, 30945.8, 2907185923.0
12273.6, 30945.8, 2907185925.0
12273.6, 30945.9, 2907185928.0
12273.6, 30945.9, 2907185930.0
12273.6, 30945.9, 2907185933.0
12273.6, 30945.8, 2907185935.0
12273.6, 30945.8, 2907185937.0
12273.6, 30945.8, 2907185940.0
12273.5, 30945.8, 2907185942.0
12273.5, 30945.9, 2907185944.0
12273.5, 30945.9, 2907185947.0
12273.5, 30945.9, 2907185949.0
12273.5, 30945.9, 2907185952.0
12273.5, 30946.0, 2907185954.0
12273.4, 30945.9, 2907185956.0
12273.4, 30945.9, 2907185959.0
12273.4, 30946.0, 2907190001.0
12273.4, 30946.0, 2907190003.0
```

B

```
1.0,     30843.5,      0.0,
12273.68,   30945.73,   2907185909.,
12273.68,   30945.74,   2907185911.,
12273.67,   30945.75,   2907185914.,
12273.66,   30945.76,   2907185916.,
12273.66,   30945.77,   2907185918.,
12273.65,   30945.78,   2907185921.,
12273.64,   30945.79,   2907185923.,
12273.63,   30945.80,   2907185925.,
12273.62,   30945.81,   2907185928.,
12273.61,   30945.82,   2907185930.,
12273.59,   30945.83,   2907185933.,
12273.58,   30945.84,   2907185935.,
12273.57,   30945.85,   2907185937.,
12273.55,   30945.86,   2907185940.,
12273.54,   30945.87,   2907185942.,
12273.52,   30945.88,   2907185944.,
12273.51,   30945.89,   2907185947.,
12273.49,   30945.90,   2907185949.,
12273.48,   30945.91,   2907185952.,
12273.47,   30945.93,   2907185954.,
12273.45,   30945.94,   2907185956.,
12273.44,   30945.95,   2907185959.,
12273.43,   30945.96,   2907190001.,
12273.42,   30945.97,   2907190003.,
```

Table C-5. Program used to calculate tow time and distance. Language: Fortran IV.

```
      PRØGRAM LENGTH (ØUTPUT,TAPE5,TAPE6)
C
C     THIS PRØGRAM CALCULATES TØW TIME AND DISTANCE (M)
C     FRØM NAVIGATIØNAL BEARINGS. THREE MEASURES ØF TØW
C     DISTANCE ARE PRØVIDED:
C      1)A TØTAL ØF THE DISTANCES CALCULATED BETWEEN
C          EACH CØNSECUTIVE NAVIGATIØNAL READING,
C      2)A TØTAL ØF THE DISTANCES CALCULATED BETWEEN
C          EVERY FIFTH NAVIGATIØNAL READING,
C      3)THE STRAIGHT-LINE DISTANCE BETWEEN THE FIRST
C          AND LAST NAVIGATIØNAL READINGS.
C     THE PERCENT DEVIATIØNS ØF MEASURES (1) AND (2) FRØM
C     MEASURE (3) ARE ALSØ GIVEN, AS IS AVERAGE SPEED (M/SEC).
C
C     THIS PRØGRAM IS PRESENTLY SET TØ DETERMINE  GEØGRAPHIC
C     PØSITIØNS USING THE LØRAN C CHAINS 593X (R1) AND 593Y (R2)
C     IN THE VICINITY ØF GEØRGES BANK (42 DEGREES W (S1); 66 DEGREES
C     N (S2)).
C
C     INITIALIZATIØN
      DIMENSIØN N(4), ZLAT(2600), ZLØN(2600)
      MN = 0
      NT = 0
      L = 0
C     K1 AND K2 CAN BE SET AT 0 SINCE
C     ØNLY LØRAN C SIGNALS ARE BEING ANALYZED
      K2 = 0
      K1 = 0
      R1 = 4H593X
      R2 = 4H593Y
C     LATITUDE (DEGREES) STARTING PØINT : S1
C     LØNGITUDE (DEGREES) STARTING PØINT : S2
      S1 = 42.
      S2 = 66.
      DID = 0.
      DIS = DID
C     THE PRØGRAM IS PRESENTLY SET UP TØ
C     READ A MAXIMUM ØF 2600 DATA VALUES PER TØW
C     (1.44 HØURS ØF RECØRDING TIME AT 2 SECØND INTERVALS)
      DØ 20 I=1,2600
         ZLAT(I) = 0.
         ZLØN(I) = 0.
   20 CØNTINUE
      N2 = 5
      NTEST = 1
      NRT = 0
      ITØW = NRT
      ISTA = 0.
      DIST = 0.
      DTPCT = 0.
      WRITE (6,130)
      WRITE (6,140)
```

.../Contd.

Table C-5 Contd...

```
C
C     READ INPUT
C
   30 CØNTINUE
      READ (5,*) A,B,C
      IF (EØF(5).NE.0) GØ TØ 120
      IC = C
      DØ 40 I=1,4
         J = 10-2*I
         N(I) = IC/10**J
         IC = IC-N(I)*10**J
   40 CØNTINUE
      DAY = N(1)
      XMØN = N(2)
      GMT = N(3)
      XMIN = N(4)
      SEC = IC
      IF (XMØN.EQ.0.) GØ TØ 80
      IF (ITØW.EQ.0) GØ TØ 30
C
C     IN TØW
C
      NRT = NRT+1
      MN = MN+1
      AØ = A
      BØ = B
      CALL LØRAN (R1,NZ1,A,K1,R2,NZ2,B,K2,XLAT,XLØN,S1,S2,L)
      IF (NRT.EQ.1) GØ TØ 60
      CALL TRACK (XLAT,XLØN,TLAT,TLØN,D)
      IF (NTEST.EQ.1) GØ TØ 50
      WRITE (6,150) A,B,C,D,DIST,XLAT,XLØN
   50 DIST = DIST+D
      TDAY = DAY
      TGMT = GMT
      TMIN = XMIN
      TSEC = SEC
   60 TLAT = XLAT
      TLØN = XLØN
      ZLAT(NRT) = XLAT
      ZLØN(NRT) = XLØN
      IF (MN.NE.5) GØ TØ 70
      NP = NRT-NZ
      IF(NRT.EQ.5)NP = 2
      PLAT = ZLAT(NP)
      PLØN = ZLØN(NP)
      CALL TRACK (XLAT,XLØN,PLAT,PLØN,DD)
      SLLAT = XLAT
      SLLØN = XLØN
      DID = DID+DD
      MN = 0
   70 IF (NRT.NE.1) GØ TØ 30
```

.../Contd.

Table C-5 Contd...

```
C
C     FIRST TØW RECØRD
C
      FLAT = XLAT
      FLØN = XLØN
      FGMT = GMT
      TIME1 = GMT*60.+XMIN+SEC/60.
      GØ TØ 30
   80 IF (SEC) 120,90,100
C
C     START ØF TØW
C
   90 NRT = 0
      ITØW = 1
      DIST = 0.
      GØ TØ 30
C
C     END ØF TØW
C
  100 TIME2 = TGMT*60.+TMIN+TSEC/60.
      IF(TGMT.LT.FGMT)TIME2 = TIME2+24.*60.
      TIME = TIME2-TIME1
      CALL TRACK (FLAT,FLØN,TLAT,TLØN,DA)
      CALL TRACK (SLLAT,SLLØN,TLAT,TLØN,DD)
      DID = DID+DD
      NT = NT+1
      DIFPCT = ((DID-DA)/DA)*100
      DIFPER = ((DIST-DA)/DA)*100
      SPEED = DID/(TIME*60.)
      ISTA = INT(A)
      WRITE (6,160) NT,ISTA,NRT,TIME,DIST,DID,DA,DIFPER,DIFPCT,SPEED
      DID = 0.
      DIST = 0.
      ITØW = 0
      NRT = 0
      ISTA = 0.
      MN = 0
      DØ 110 I=1,2600
         ZLAT(I) = 0.
         ZLØN(I) = 0.
  110 CØNTINUE
      GØ TØ 30
  120 STØP
C
C
C
  130 FØRMAT (/54H #  STN  NRT  TØW   DIST. EACH  DIST. EACH   ST. LINE
     $,4X,26H%DIFF. %DIFF.    AVG SPEED)
  140 FØRMAT (14X,4HTIME,6X,43HRECØRD  5TH RECØRD    DISTANCE     1-ST.
     $5-,17HST.       ( M/SEC)
  150 FØRMAT (2F12.1,F16.2,2F13.3,2F18.10)
  160 FØRMAT (I3,1X,I3,1X,I4,1X,F5.2,3F12.2,4X,2(F6.2,1X),6X,F5.3)
      END
```

.../Contd.

Table C-5 Contd...

```
      SUBRØUTINE LØRAN (RATE1,NZØN1,READ1,K1,RATE2,NZØN2,READ2,K2,XLAT,
     $XLØN,START1,START2,LIST,IER)
C
C     THIS SUBRØUTINE PRØVIDES AN ITERATIVE SØLUTIØN TØ CØNVERT
C     HYPERBØLIC NAVIGATIØNAL READINGS TØ GEØGRAPHIC PØSITIØN,
C     TAKING INTØ CØNSIDERATIØN THE ELLIPTICITY ØF THE EARTH.
C     IT IS PRESENTLY SPECIFIC TØ NØRTHWEST ATLANTIC LØRAN C, DECCA
C     AND LØRAN A READINGS,BUT BY MØDIFYING SUBRØUTINE CHAIN, IT CAN
C     BE ADAPTED TØ ANY AREA. REQUIRED CHANGES SHØULD BE ØBTAINABLE
C     THRØUGH THE CØAST GUARD ØR ANY ØCEANØGRAPHIC INSTITUTIØN.
C
C     RATE1,2 : FIRST AND SECØND RATES (4 CØLS. EACH)
C     NZØN1,2 : FIRST AND SECØND ZØNES FØR DECCA (1 CØL. EACH)
C     READ1,2 : READINGS FØR THE FIRST AND SECØND RATES (7 CØLS. EACH;
C               INCLUDE DECIMAL)
C     K1,2 : ZØNE INDICATØRS FØR RESPECTIVE DECCA READINGS (1 CØL.
C               EACH)
C     START1 : INITIAL ESTIMATE ØF LATITUDE (DEGREES)
C     START2 : INITIAL ESTIMATE ØF LØNGITUDE (DEGREES)
C
      CØMMØN /Y/ PI,AE,FL
      DIMENSIØN PA(2), CR(2), DR(2), PØSN(2,2), DIST(100), LSET(100)
      DIMENSIØN HYP(2), X(2)
      DIMENSIØN NTIM(100), SPD(100), R(4), A(4), B(4)
      DIMENSIØN PØS(6,2), XLIST(2), WIDTH(2), CØUNT(2), BASE(2)
      XLØN = 0.
      XLAT = XLØN
      M = 0
      INT = 5
      TØT = 9.0
C     AE : RADIUS ØF EARTH
C     FL : FLATTENING ØF REFERENCE ELLIPSØID
      AE = 6378206.4
      FL = 0.33900753E-02
      PI = 4.0*ATAN(1.0E0)
      GØ TØ 30
   10 WRITE (6,120) RATE1,NZØN1,READ1,RATE2,NZØN2,READ2
      IER = 7
      GØ TØ 110
   20 WRITE (6,140) RATE1,RATE2
      GØ TØ 110
   30 PØS(5,1) = START1*PI/180.0
      PØS(5,2) = START2*PI/180.0
C
C
      CALL CHAIN (RATE1,NZØN1,READ1,K1,PØS(2,1),PØS(2,2),PØS(1,1),PØS(1,
     $2),PA(1),CØUNT(1),IER)
      IF (IER.NE.0) GØ TØ 20
      CALL CHAIN (RATE2,NZØN2,READ2,K2,PØS(4,1),PØS(4,2),PØS(3,1),PØS(3,
     $2),PA(2),CØUNT(2),IER)
      IF (IER.NE.0) GØ TØ 20
      CALL SPHRD (PØS(1,1),PØS(1,2),PØS(2,1),PØS(2,2),BASE(1),A1,A2)
      CALL SPHRD (PØS(3,1),PØS(3,2),PØS(4,1),PØS(4,2),BASE(2),A1,A2)
```

.../Contd.

Table C-5 Contd...

```
      DØ 40 I=1,2
         WIDTH(I) = BASE(I)/CØUNT(I)
   40 CØNTINUE
      J = 5
      ITER = 0
   50 CØNTINUE
      ITER = ITER+1
      IF (ITER.GT.40) GØ TØ 10
      DØ 60 I=1,4
         CALL SPHRD (PØS(J,1),PØS(J,2),PØS(I,1),PØS(I,2),R(I),A(I),B(I))
   60 CØNTINUE
      R(2) = R(1)-R(2)
      R(3) = R(3)-R(4)
      A(2) = A(1)-A(2)
      B(2) = B(1)-B(2)
      A(3) = A(3)-A(4)
      B(3) = B(3)-B(4)
      DØ 70 I=1,2
         CR(I) = R(I+1)+BASE(I)
         A(I) = A(I+1)
         B(I) = B(I+1)
         HYP(I) = PA(I)*WIDTH(I)*2.0
         DR(I) = HYP(I)-CR(I)
   70 CØNTINUE
      X1 = DR(1)
      X2 = DR(2)
      TEST = ABS(X1)+ABS(X2)
      RLAX = 0.95
      DET = A(1)*B(2)-A(2)*B(1)
      DØ 80 I=1,2
         A(I) = A(I)/DET
         B(I) = B(I)/DET
   80 CØNTINUE
      TMP = A(1)
      A(1) = B(2)
      B(2) = TMP
      A(2) = -A(2)
      B(1) = -B(1)
      DØ 90 I=1,2
         PØS(5,I) = PØS(5,I)+A(I)*RLAX*DR(1)+B(I)*RLAX*DR(2)
   90 CØNTINUE
      IF (TEST.GT.3.0) GØ TØ 50
      X(1) = PØS(5,1)
      X(2) = PØS(5,2)
      DØ 100 I=1,2
         X(I) = X(I)*180./PI
         IF(X(I).LT.0.)X(I) = X(I)+360.
         IF(X(I).GE.360.)X(I) = X(I)-360.
  100 CØNTINUE
      XLAT = X(1)
      XLØN = X(2)
      IF (LIST.EQ.0) GØ TØ 110
```

.../Contd.

Table C-5 Contd...

```
          CALL LALØ (XLAT,XLØN,LAT,LØN)
          WRITE (6,130) RATE1,NZØN1,READ1,RATE2,NZØN2,READ2,LAT,LØN
      110 RETURN
    C
    C
    C
      120 FØRMAT (24H UNABLE TØ PLØT PØSITIØN,2X,A4,2X,A1,F10.2,5X,A4,2X,A1,
         $F10.2)
      130 FØRMAT (1X,A4,3X,A1,F10.2,2X,A4,3X,A1,F10.2,2I10)
      140 FØRMAT (17H INVALID RATE -  ,A4,4H ØR ,A4)
          END
          SUBRØUTINE CHAIN (RATE,NZØNE,READ,K,XLATM,XLØNM,XLATS,XLØNS,PA,
         $CØUNT,IER)
          CØMMØN /Y/ PI
          DIMENSIØN ALATM(38), ALØNM(38), ALATS(38), ALØNS(38), ARATE(38)
          DIMENSIØN XCØUNT(38), IDELTA(38), IZØNS(11), ALØRC(13)
          DATA ALATM/47.73997778,46.91839444,49.34356667,47.63251389,45.
         $73648889,45.65402778,43.70289167,44.96462222,47.84863333,50.
         $17985833,50.15224444,45.21485833,45.21485833,41.24993333,59.
         $98862222,48.69601667,41.24984722,41.24898056,40.47265556,76.
         $31934722,69.44951111,2*46.80755556,4*30.99409444,4*42.71405556,4*
         $34.06276667,3*59.98810833/
          DATA ALØNM/52.73705000,55.37942222,54.87013056,59.23921389,61.
         $90060278,64.93010833,65.24106111,62.14964722,64.69019722,61.
         $81303611,66.61764444,2*61.175650,69.97232500,44.65243333,53.088400
         $,69.97271667,69.97540556,74.01227778,69.35860000,54.24124167,2*67.
         $92714167,4*85.16925,4*76.82607222,4*77.91298889,3*45.17429722/
          DATA ALATS/3*48.35371389,2*47.35723056,3*44.56715278,3*49.85835278
         $,47.56494444,43.45929167,43.45927222,52.24737222,52.24758333,43.
         $91811111,35.24053611,41.24898056,2*70.52873056,41.25331389,46.
         $77560556,30.72583889,26.53194722,27.03291389,34.06278889,46.
         $80755556,41.25331389,34.06278889,39.85209444,27.03289444,41.
         $25328889,46.77558056,39.85207222,64.90736667,62.29989167,46.
         $77558056/
          DATA ALØNS/3*54.16905,2*61.92671667,3*64.26818889,3*64.44477222,59
         $.16408056,65.47120278,65.46971111,55.61104444,55.61200000,69.
         $26095000,75.52717500,69.97540556,68.29701944,68.29701944,69.
         $97752500,53.17448889,90.82877778,97.83335833,80.11486667,77.
         $91298889,67.92714167,69.97752500,77.91298889,87.48670556,80.
         $11486667,69.97752500,53.17448889,87.48670556,23.92270833,07.
         $07408611,53.17448889/
          DATA XCØUNT/286.1079,311.4186,343.2388,465.6644,307.5821,299.7552,
         $211.4852,494.5979,513.6260,329.8720,455.3010,2026.1392,2630.4552,
         $2967.5415,7314.1422,2894.5656,2015.9625,5504.3892,2342.9140,4318.
         $2113,3663.6454,4263.76,7509.86,3619.08,8886.76,4403.76,5085.44,
         $5594.40,3939.86,6443.30,6324.12,5391.02,7082.62,16779.30,7121.42,
         $8136.06,13607.52,10424.38/
          DATA IDELTA/0,30,50,0,30,0,30,50,0,30,50,4*1000,950,5*1000,11000,
         $25000,11000,23000,43000,59000,11000,25000,39000,54000,11000,49000,
         $28000,65000,11000,21000,43000/
          DATA IZØNS/24,18,30,24,18,24,18,30,24,18,30/
```

.../Contd.

Table C-5 Contd...

```
      DATA ARATE/4H2R  ,4H2G  ,4H2P  ,4H6R  ,4H6G  ,4H7R  ,4H7G  ,4H7P
     $,4H9R  ,4H9G  ,4H9P  ,4H1H1 ,4H1H2 ,4H1H3 ,4H1L2 ,4H1L3 ,4H1H7 ,
     $4H3H4 ,4H3H5 ,4H2S6 ,4H2S7 ,4HSH7X,4HSH7Y,4HSL2W,4HSL2X,4HSL2Y,
     $4HSL2Z,4HSS4W,4HSS4X,4HSS4Y,4HSS4Z,4HSS7W,4HSS7Y,4HSS7X,4HSS7Z,
     $4HSL7W,4HSL7X,4HSL7Z/
      DATA ALØRC/4H593X,4H593Y,4H996W,4H996X,4H996Y,4H996Z,4H993W,4H993Y
     $,4H993X,4H993Z,4H793W,4H793X,4H793Z/
      RAD = PI/180.0
      IER = 0
      DØ 10 I=1,38
         IF (RATE.EQ.ARATE(I)) GØ TØ 40
   10 CØNTINUE
      DØ 20 I=1,13
         IF (RATE.EQ.ALØRC(I)) GØ TØ 30
   20 CØNTINUE
      IER = 1
      GØ TØ 60
   30 IF(I.GT.2)I = I+4
      I = I+21
   40 XLATM = ALATM(I)*RAD
      XLØNM = ALØNM(I)*RAD
      XLATS = ALATS(I)*RAD
      XLØNS = ALØNS(I)*RAD
      DELTA = IDELTA(I)
      CØUNT = XCØUNT(I)
      IF (K.GT.0) GØ TØ 50
      PA = READ-DELTA
      GØ TØ 60
   50 CALL ZØNE (K,NZØNE,KZØNE)
      PA = KZØNE*IZØNS(I)+READ-DELTA
   60 RETURN
      END
C     ANDØYER-LAMBERT FØR DISTANCE, AZIMUTHS AND DIRFF. CØEFF.
C
      SUBRØUTINE SPHRD (PHP,FLP,PHS,FLS,RANGE,A1,A2)
      CØMMØN /Y/ PI,AE,G
      DLØN = FLS-FLP
      SL = SIN(DLØN)
      CL = CØS(DLØN)
      SP = SIN(PHP)
      CP = CØS(PHP)
      SS = SIN(PHS)
      CS = CØS(PHS)
      C = SS*SP+CS*CP*CL
      S = SQRT(1.0-C*C)
      U = ATAN(S/C)
      CØSU = C
C     ELLIPSØID CØRRECTIØNS
      DV = (U+3.0*S)/(1.0-C)
      DV = DV*(SP-SS)*(SP-SS)
      DW = (U-3.0*S)/(1.0+C)
      DW = DW*(SP+SS)*(SP+SS)
      RANGE = AE*(U-0.25*G*(DV+DW))
```

Table C-5 Contd...

```
      A1 = -CP*SS+SP*CS*CL
      A1 = AE*A1/S
      A2 = -AE*CP*CS*SL/S
   10 RETURN
      END
      SUBRØUTINE ZØNE (K,LPHA,NUMER)
      INTEGER A,B,C,D,E,F,G,H,I,J
      DATA A,B,C,D,E/1HA,1HB,1HC,1HD,1HE/
      DATA F,G,H,I,J/1HF,1HG,1HH,1HI,1HJ/
      NUMER = 0.
      L = (K-1)*10
      IF(LPHA.EQ.A)NUMER = L
      IF(LPHA.EQ.B)NUMER = L+1
      IF(LPHA.EQ.C)NUMER = L+2
      IF(LPHA.EQ.D)NUMER = L+3
      IF(LPHA.EQ.E)NUMER = L+4
      IF(LPHA.EQ.F)NUMER = L+5
      IF(LPHA.EQ.G)NUMER = L+6
      IF(LPHA.EQ.H)NUMER = L+7
      IF(LPHA.EQ.I)NUMER = L+8
      IF(LPHA.EQ.J)NUMER = L+9
      RETURN
      END
      SUBRØUTINE TRACK (XLA1,XLØ1,XLA2,XLØ2,DIST)
C
C     THIS RØUTINE CALCULATES THE DISTANCE IN METRES BETWEEN TWØ
C     GEØGRAPHICAL PØINTS.
C
      DATA PI/3.14159265359/,SEC180/648000./
C
C     SEC180 - NØ. ØF SECØNDS IN 180 DEGREES
C
C     CVR - CØNVERSIØN FACTØR FRØM SECØNDS TØ RADIANS
      CVR = PI/SEC180
C
      WRK1 = XLA1*3600.
      WRK2 = XLA2*3600.
      FLA = RADE(WRK1/3600.)*CVR
      FLØ = FLA*CØS(WRK1*CVR)
C
C     FLA - FACTØR TØ CØNVERT SECØNDS ØF ARC TØ METERS FØR LATITUDE
C     FLØ -    "     "    "         "    "    "   "     "      "  LØNGITUDE
C     W - NØ. ØF SECØNDS BETWEEN LATITUDE ØR LØNGITUDE
C
      W = WRK1-WRK2
      DSAM = ABS(W)*FLA
      WRK1 = XLØ1*3600.
      WRK2 = XLØ2*3600.
      W = WRK1-WRK2
      DSØM = ABS(W)*FLØ
      DIST = SQRT((DSØM**2)+(DSAM**2))
      RETURN
      END
```

.../Contd.

Table C-5 Contd...

```
      FUNCTIØN RADE (XLAT)
C     THIS FUNCTIØN CALCULATES THE RADIUS ØF THE EARTH
C     IN METERS AT A GIVEN LATITUDE IN DECIMAL DEGREES
C
C     THE EQUATØRIAL RADIUS ØF THE EARTH IS :
      RADEQ = 6378206.4
C     DØUBLE THE LATITUDE IN RADIANS IS :
      DLATR = XLAT*3.14159265359/90
      RADE = 0.998327073+0.001676438*CØS(DLATR)
      RADE = RADE-0.000003519*CØS(2*DLATR)
      RADE = RADE+0.000000008*CØS(3*DLATR)
      RADE = RADE*RADEQ
      RETURN
      END
      SUBRØUTINE LALØ (XLAT,XLØNG,LAT,LØNG)
C
C     THIS RØUTINE CØNVERTS LAT & LØNG FRØM FLØATING PT.
C     TØ INTEGER (DEG-MIN-SEC)
C
      HALFSEC = 1./7200.
C
      R = XLAT
      DØ 30 I=1,2
         R = R+HALFSEC
         ID = R
         RD = ID
         X = (R-RD)*60.
         IM = X
         XM = IM
         XS = (X-XM)*60.
         IS = XS
         IF (IS.LT.60) GØ TØ 10
         IS = IS-60
         IM = IM+1
         IF (IM.LT.60) GØ TØ 10
         IM = IM-60
         ID = ID+1
   10    CØNTINUE
         L = ID*10000+IM*100+IS
         IF (I.EQ.2) GØ TØ 20
         LAT = L
         R = XLØNG
         GØ TØ 30
   20    LØNG = L
   30 CØNTINUE
      RETURN
      END
```

Table C-6. Sample output of analysis program (Table C-5)

#	STN	NRT	TØW TIME	DIST. EACH RECØRD	DIST. EACH 5TH RECØRD	ST. LINE DISTANCE	%DIFF. 1-ST.	%DIFF. 5-ST.	AVG SPEED (M/SEC)
1	3	203	8.00	1476.11	1456.46	1454.24	1.50	.15	3.034
2	4	158	6.20	821.06	801.60	794.43	3.35	.90	2.155
3	5	141	5.53	803.96	794.67	794.12	1.24	.07	2.394
4	6	116	4.55	388.63	369.33	360.17	7.90	2.54	1.353
5	7	115	14.57	1669.03	1598.11	1175.38	42.00	35.97	1.828
6	8	183	7.18	701.91	682.73	659.82	6.38	3.47	1.584
7	9	212	9.43	1718.69	1701.52	1686.63	1.90	.88	3.006
8	10	235	9.25	1127.83	1109.50	1099.32	2.59	.93	1.999
9	11	253	11.05	1120.47	1090.45	1079.33	3.81	1.03	1.645
10	12	212	8.33	759.96	728.25	715.48	6.22	1.78	1.456
11	13	185	8.47	838.64	816.04	815.66	2.82	.05	1.606