United States

Department of the Interior

Geologic Survey

Menlo Park, California 94025

SQUASH/MULTICS

A Computer Program to be Used in Conjunction with HYPOELLIPSE to Generate an Augmented Phase Data Archive

by

John C. Lahr

Open-file Report

No.

80-375

CONTENTS

Introduction

The program SQUASH was written as a "post processor" for the earthquake location program HYPOELLIPSE (Lahr, 1979) as it has been implemented on the USGS MULTICS computer. In the past we have maintained two machine readable archives of earthquake related data. The first is PHASE data file which contains, for each earthquake, the original seismogram readings (e.g., arrival time and coda length) for each seismic station used in the solution. The second was the SUMMARY data file with one 80 column entry for each earthquake containing data such as epicenter, depth, origin time, magnitude, and solution quality. In addition, a print file was generated with specific information for each station used, such as the epicentral distance, arrival time residual, azimuth from epicenter to station, angle of incidence of the ray, and magnitude. This archival procedure made it extremely tedious to make many types of studies. For example, a plot of coda length versus distance for earthquakes within some magnitude and/or regional bounds would require checking the printout for particular events and plotting the data by hand or keying the appropriate data into a computer.

Using HYPOELLIPSE followed by SQUASH should facilitate many studies because one augmented data file is created. For each earthquake the EXTENDED-PHASE FILE (EPF) contains one entry for each station used, followed by the "summary card" entry. The EPF is formatted in such a way that it can be used as an input PHASE data file for a subsequent run of HYPOELLIPSE. On a subsequent run, the starting trial location for the hypocenter is set to the location given on the summary card from the previous run. This will significantly reduce the computation required to converge on a new hypocenter, as compared with starting at the closest station again. The EPF has a 104 column entry for each station and a 91 column "summary card" entry. Since more than 80 columns are used, the EPF cannot be stored conveniently on cards.

How to Use SQUASH

SQUASH operates on the file60 generated by HYPOELLIPSE/MULTICS when PUNCH OPTION is set equal to 2 and the first entry after ARRIVAL TIMES NEXT is SAVE in columns 1-4. The following illustration shows how SQUASH would be used after file60 was generated by HYPOELLIPSE.

```
io attach file05 vfile_ file60
io open file05 si
io attach file50 vfile_ EPF
io open file50 so
> udd > Akss > library > squash
```

Format of EXTENDED-PHASE FILE (EPF)

A. Station Entry

Items	Col. Nos.	Format for Reading
Station name	1 - 4	A4
Any 2 alphanumneric symbols to describe P-phases.	5 - 6	A2
First motion direction of P-arrival C or U Compression D Dilatation + Poor Compression - Poor dilatation N-Noisy Blank-not readable	7	A1
P-phase weight code 0 or blank full weight 1 2 Partial weight 3 Weight = ((4.-Code)/4)**TEST(36) 4, 5, 6, 7, 8 no weight	8	F1.0
If the P-phase is a secondary arrival refracted along the bottom of the Ith layer, punch the value of I here. If event is in the (I+1)th layer a direct wave calculation will be made. Below that, the weight is reset to zero.	9	I1
Year, month, day, hour, minute	10 - 19	I10
Seconds of P-arrival	20 - 24	F5.2
Distance (km)	25 - 28	F4.1
AZM - Azimuth from epicenter to station.	29 - 31	F3.0
Seconds of S arrival	32 - 36	F5.2
S-remark	37 - 39	A3
S-phase weight code	40	F1.0
AIN - Angle of ray leaving hypocenter.	41 - 43	F3.0
Maximum peak-to-peak amplitude in mm.	44 - 47	F4.0
Period of maximum amplitude in sec. If left blank, the standard period as specified in the station list will be used.	48 - 50	F3.2

Format of EPF (continued)

A. Station Entry

Items	Col. Nos.	Format for Reading
P travel time computed	51 - 54	F4.2
P weight, final	55 - 57	F3.2
D, B, M or * weight code (See C.6 for definition)	58	A1
Peak-to-peak amplitude of 10uv calibration in mm. If this is blank then the standard calibration from the station cards is used.	59 - 62	F4.1
Any remarks except CAL.	63 - 65	A3
Time correction in seconds.	66 - 70	F5.2
F-P time interval in sec. for FMAG	71 - 75	F5.0
P:RES - Residual of P-arrival in seconds.	76 - 80	F5.2
S weight, final	81 - 83	F3.2
D, B, M or * weight code	84	A1
S:RES - Residual of S-arrival in seconds.	85 - 89	F5.2
P delay.	90 - 92	F3.2
S delay.	93 - 95	F3.2
P elevation delay.	96 - 98	F3.2
System response code.	99 - 100	I2
XMAG	101 - 102	F2.1
FMAG	103 - 104	F2.1

B. Summary Entry

Items	Col. Nos.	Format for Reading
KDATE - year, month, day Origin time	1 - 6	I6
KHRMN - hour, minute,	7 - 10	I4
seconds	11 - 14	F4.2
LAT degrees	15 - 16	I2
N or S	17	A1
LAT minutes	18 - 21	F4.2
LON degrees	22 - 24	I3
E or W	25	A1
LON minutes	26 - 29	F4.2
DEPTH (km)	30 - 34	F5.2
MAGNITUDE preferred	35 - 36	F2.1
NO-Number of P, S, and S-P readings used in the solution	37 - 39	I3
GAP - Largest azimuthal separation in degrees between stations as seen from the epicenter (deg.).	40 - 42	I3
D3 - Distance to third most distant station (km).	43 - 45	F3.0
RMS (sec)	46 - 49	F4.2

Format of EPF (continued)

B. Summary Entry (continued)

SIZE AND ORIENTATION OF ERROR ELLIPSOID:

Items	Col. Nos.	Format for Reading
Azimuth (deg.)	50 - 52	I3
Dip (deg.)	53 - 54	I2
SE - length of ellipsoid axis	55 - 58	F4.2
Azimuth (deg.)	59 - 61	I3
Dip (deg.)	62 - 63	I2
SE - length of ellipsoid axis	64 - 67	F4.2
Average XMAG	68 - 69	F2.1
Average FMAG	70 - 71	F2.1
SE - length of third ellipsoid axis	73 - 76	F4.2
Quality - either error ellipsoid quality or HYPO quality depending upon Quality Option Card. See Section B.3	77	A1
SETNO - Identification symbol as punched in column 80 of HEADER CARD. See Section B.1. If MAGNITUDE OPTION 3 is used, this will be F or X to indicate which type of magnitude was preferred.	78	A1
NSWT - Number of S-phase arrivals used in solution.	79 - 80	I2
/	81	A1
First 8 characters of INSTRUCTION card.	82 - 89	2A4
S data indicator from INSTRUCTION card.	90	I1
Fixed location indicator from INSTRUCTION card.	91	I1

References

Lahr, J. C., 1979, HYPOELLIPSE/MULTICS: A computer program for determining local earthquake hypocentral parameters, magnitude, and first motion pattern, U.S. Geological Survey, Open-File Report 80-59, 59 p.

```
COMPILATION LISTING OF squash (>user_dir_dir>Akss>JLahr>hypoellipse>squash.fortran)

Compiled by: Multics New Fortran Compiler, Release 5a
Compiled on: 09/05/79  1650.7 pdt Wed
     Options: table card map

                                        Main Program

 1 c      PROGRAM TO COMPACT EARTHQUAKE SUMMARY INFORMATION
 2 c         FOR USE WITH HYPOELLIPSE PUNCH OPTION 2 AND SAVE
 3        dimension name(150),ap(150),bp(150),cp(150),dp(150),ep(150)
 4        dimension fp(150),icard(150)
 5        integer flag
 6        character*1 slash,itest
 7        character*3 extra2
 8        character*4 name,ap,bp,name1
 9        character*7 ep,extra1
10        character*9 dp
11        character*12 cp,extra3
12        character*13 fp
13        character*29 extra4
14        character*81 icard,jcard
15        character*91 incard
16        rewind 05
17        nevents = 0
18     10 nevents = nevents + 1
19     20 np = 1
20     30 read(5,35,end=1000) jcard
21     35 format(a81)
22     36 decode(jcard,40) name(np),ap(np),bp(np),cp(np),
23       1 dp(np),ep(np),fp(np),itest
24        icard(np) = jcard
25     40 format(3a4,a12,7x,a9,3x,a7,12x,a13,5x,a1)
26        if(name(np) .ne. 4h $$$) go to 45
27        np = np - 1
28        write(6,44) name(np),ap(np),bp(np),cp(np)
29     44 format(" out of sync at card: ",3a4,a12," so stop")
30        go to 1000
31     45 if(itest .eq. 1h/) go to 50
32        if(bp(np) .eq. 4h     ) go to 50
33        np = np + 1
34        go to 30
35     50 np = np - 1
36        if(np .gt. 0) go to 70
37        if(name(i) .eq. 4hTAPE) go to 20
38        write(6,60) (name(i),ap(i),bp(i),cp(i),i=1,2)
39     60 format(" extra inst. card found:",3a4,a12,/," 2nd cd of previous
40       1event is:",3a4,a12)
41        go to 20
42     70 read(5,80) incard,slash
43     80 format(a91,t81,a1)
44        if(slash .eq. 1h/) go to 82
45        do 83 i = 1,np + 1
46        write(50,35) icard(i)
47     83 continue
48        nevents = nevents + 1
49        np = 1
50        decode(incard,84) jcard
51     84 format(a81)
52        go to 36
```

```
 82 flag = 0
    npp = 1
 85 read(5,90) name1,extra1,extra2,extra3,extra4
 90 format(a4,20x,a7,9x,a3,7x,a12,13x,a29)
    if(flag .eq. 1) go to 130
    if(name1 .eq. 4h $$$ ) go to 150
100 if(name1 .eq. name(npp)) go to 110
    write(50,40) name(npp),ap(npp),bp(npp),cp(npp),dp(npp),ep(npp),
   1 fp(npp)
    npp = npp + 1
    if(npp .le. np) go to 100
    write(6,105) name1
105 format(" error, can not find ",a4," so stop")
    go to 1000
110 write(50,120) name(npp),ap(npp),bp(npp),cp(npp),extra1,dp(npp),
   1 extra2,ep(npp),extra3,fp(npp),extra4
120 format(3a4,a12,a7,a9,a3,a7,a12,a13,a29)
    npp = npp + 1
    if(npp .le. np) go to 85
    flag = 1
    go to 85
130 if(name1 .ne. 4h $$$ ) go to 138
    write(50,135) incard
135 format(a91)
    go to 10
138 write(6,140) name1
140 format(" extra card in extra file: ",a4," so stop")
    go to 1000
150 do 160 i = npp,np
    write(50,40) name(npp),ap(npp),bp(npp),cp(npp),dp(npp),
   1 ep(npp),fp(npp)
160 continue
    go to 10
1000 nevents = nevents - 1
    write(6,1010) nevents
1010 format(1x,i10," events processed")
1020 stop
    end
```

NAMES USED IN THIS PROGRAM UNIT

NAME	TYPE OF NAME	LOC	STORAGE	ATTRIBUTES AND REFERENCES
ap		000406	automatic	character(4) array(150) ref 3 8 22 28 38 60 67 82
bp		000634	automatic	character(4) array(150) ref 3 8 22 28 32 38 60 67 82
cp		001062	automatic	character(12) array(150) ref 3 11 22 28 38 60 67 82
dp		001764	automatic	character(9) array(150) ref 3 10 22 60 67 82
ep		002666	automatic	character(7) array(150) ref 3 9 22 60 67 82
extra1		012615	automatic	character(7) ref 9 55 67
extra2		012613	automatic	character(3) ref 7 55 67
extra3		012617	automatic	character(12) ref 11 55 67
extra4		012622	automatic	character(29) ref 13 55 67
flag		012610	automatic	integer ref 5 53 57 72
fp		003342	automatic	character(13) array(150) ref 4 12 22 60 67 82
i		012710	automatic	integer ref 37 38 38 38 38 38 45 46 81
icard		004472	automatic	character(81) array(150) ref 4 14 24 46
incard		012657	automatic	character(91) ref 15 42 50 75
itest		012612	automatic	character(1) ref 6 22 31
jcard		012632	automatic	character(81) ref 14 20 22 24 50
main_	entry point	000001	constant	on line 3
name		000160	automatic	character(4) array(150) ref 3 8 22 26 28 37 38 59 60 67 82
name1		012614	automatic	character(4) ref 8 55 58 59 64 74 78
nevents		012706	automatic	integer ref 17 18 18 48 48 86 86 87
np		012707	automatic	integer ref 19 22 22 22 22 22 22 22 22 24 26 27 27 28 28 28 28 32 33 33 35 35 36 45 49 63 71 81
npp		012711	automatic	integer ref 54 59 60 60 60 60 60 60 60 62 62 63 67 67 67 67 67 67 67 70 70 71 81 82 82 82 82 82 82 82
slash		012611	automatic	character(1) ref 6 42 44

LOC	LABEL	TYPE	LINE	REFERENCES
000025	10	executable	18	used in transfer ref 77 85
000030	20	executable	19	used in transfer ref 37 41
000032	30	executable	20	used in transfer ref 34
	35	format	21	ref 20 46
000053	36	executable	22	used in transfer ref 52
	40	format	25	ref 22 60 82
	44	format	29	ref 28
000240	45	executable	31	used in transfer ref 26
000260	50	executable	35	used in transfer ref 31 32
	60	format	39	ref 38
000346	70	executable	42	used in transfer ref 36
	80	format	43	ref 42
000465	82	executable	53	used in transfer ref 44
000430	83	executable	47	ref 45
	84	format	51	ref 50
000471	85	executable	55	used in transfer ref 71 73
	90	format	56	ref 55
000545	100	executable	59	used in transfer ref 63
	105	format	65	ref 64
000674	110	executable	67	used in transfer ref 59
	120	format	69	ref 67
001023	130	executable	74	used in transfer ref 57
	135	format	76	ref 75
001050	138	executable	78	used in transfer ref 74
	140	format	79	ref 78
001070	150	executable	81	used in transfer ref 58

```
001166        160   executable      84              ref 81
001173       1000   executable      86              used in transfer ref 20 30 66 80
             1010   format          88              ref 87
001217       1020   executable      89
```

LINE	LOC	LINE	LOC	LINE	LOC	LINE	LOC	LINE	LOC	LINE	LOC	LINE	LOC
0	000000	16	000016	17	000023	18	000025	19	000030	20	000032	22	000053
24	000156	26	000164	27	000172	28	000175	30	000237	31	000240	32	000245
33	000254	34	000257	35	000260	36	000263	37	000266	38	000275	41	000345
42	000346	44	000372	45	000377	46	000406	47	000430	48	000434	49	000437
50	000441	52	000464	53	000465	54	000467	55	000471	57	000534	58	000540
59	000545	60	000554	62	000646	63	000651	64	000654	66	000673	67	000674
70	001012	71	001015	72	001020	73	001022	74	001023	75	001030	77	001047
78	001050	80	001067	81	001070	82	001074	84	001166	85	001172	86	001173
87	001200	89	001217										

```
OBJECT SEGMENT SUMMARY

STORAGE REQUIREMENTS FOR THIS PROGRAM

          Object     Text      Link      Symbol     Defs      Static
Start          0        0      1240        1500     1450        1250
Length      2576     1240       210        1061       27         176

Stack frame is 5584 (decimal) words.


ENTRY POINT          LOC       ATTRIBUTES

main_                000001    on line 3

NO EXTERNAL REFERENCES

NO COMMON BLOCKS
```