The Shallow Velocity Structure of Broken Ridge

by

Kevin R. MacKenzie

Final Report of Research Agreement

per
Letter Agreement #4
under NSF Proposal OCE 85-16918

with
Columbia University
Lamont-Doherty Geological Observatory

Institute for Geophysics
University of Texas at Austin
8701 Mopac Blvd.
Austin, TX 78759-8345

UTIG Technical Report No. 102

April 20, 1989

ABSTRACT

In September 1986, J. Weissel, G. Mountain and G. Karner conducted a high-resolution single channel seismic (SCS) survey of the central Broken Ridge, an aseismic ridge located in the Eastern Indian Ocean. As an adjunct to this survey, we performed a high resolution seismic refraction study over the same study area. The seismic source consisted of twin 80 cu. in. water guns fired at a nominal shot spacing of 50 m. Both the telemetered SSQ-57A sonobuoy data and the seismic streamer outputs were sampled at 1 msec intervals. We deployed sonobuoys along dip and strike lines of the ~3000 km of SCS profiling over the principal 120 x 230 km study area.

SCS profiles along dip lines of the ridge show the thick northward dipping carbonate and chert sequence, truncated near the crest of the ridge by a horizontal angular unconformity. The unconformity is overlain by a broad lenzoid unit along the summit of the ridge. Pelagic sediments cap the ridge, draping down the northward slope where they lie conformably on the carbonate sequence. Drilling at DSDP Site 255, at the very crest of the ridge where the angular unconformity nearly outcrops, sampled limestones of Santonian age beneath a sequence of Middle Eocene detrital sands and gravels containing reworked Early Eocene materials. It is apparent from Line 20 of the SCS data that the limestones sampled lie well down from the inferred top of the carbonate and chert sequence and thus provide little control on the minimum age of the sequence or the onset of the rifting event which separated Broken Ridge from its conjugate, the Kerguelen Plateau.

The sonobuoy data exhibit good delineation of precritical reflections and wide angle arrivals from the Neogene-to-Cretaceous sedimentary sequence. Total sub-seafloor penetration approaches 2.5 km although arrivals from basaltic basement are generally very weak if present. There is very good correlation of most of the precritical reflection events with the mappable horizons observed in the SCS data. Ray-trace modeling of both the wide-angle and precritical arrivals for 22 sonobuoys resolve layer thicknesses on scales of 50-75 m. This is sufficient to map the pinch-out of the lagoonal-like deposits noted by Weissel at the crest of the ridge. Velocities in the carbonate and chert sequence along the unconformity are observed to increase to the south as stratigraphically deeper sections of the carbonate are exposed. This suggests that the higher velocities may represent increased diagenesis rather than compaction and dewatering.

Weissel and others have proposed four drilling sites, BR-1 to BR-4, for ODP Leg 121 to sample the angular unconformity along a N-S transect so as to identify and date the pre-, syn- and post-rifting stratigraphic events relating to the breakup of the Broken Ridge and the Kerguelen Plateau. The seismic structure derived here indicate that this is feasible with the planned series of nominally 400 m deep holes.

INTRODUCTION

The Broken Ridge represents one of the more enigmatic structures in the eastern Indian Ocean. With its crest oriented approximately E-W, this oceanic plateau resembles a tilt block structure in cross section. Its very steep southern scarp faces the Kerguelen Plateau, which is situated symmetrically on the opposite side of the Southeast Indian Ocean Ridge. The growing body of evidence supports the conjugal relationship between these two oceanic plateaus which were rifted apart by the Southeast Indian Ocean Ridge approximately 45 Ma.

J. Weissel, G. Mountain and G. Karner proposed to investigate the stratigraphy of the Broken Ridge to investigate processes associated with rifting event which separated it from the Kerguelen Plateau. In particular, they hoped to determine whether the uplift of the Broken Ridge should be attributed to "active" rifting (ie., response to aesthenospheric processes) or "passive" (ie., flexural response to unloading of the plate edge). To establish the detailed stratigraphy of the ridge and to select sites for a series of holes to be drilled during Leg 121 of the Ocean Drilling Program, they conducted a high resolution single channel seismic (SCS) survey of the central Broken Ridge.

A thorough understanding of the seismic velocity structure to a subbottom depth of at least one kilometer is critical to the interpretation of the high resolution SCS data for Broken Ridge and to establish feasible drilling targets. Therefore, we performed a seismic refraction survey in conjunction with the high resolution SCS studies to provide both velocity and depth information for key reflection events.

DATA ACQUISITION

The seismic refraction data were acquired during the SCS survey of the Broken Ridge conducted on R/V CONRAD cruise RC2708 in September of 1986. Free-floating, disposable SSQ-57A military sonobuoys were used as receivers. The sensor consisted of a hydrophone suspended at 122 m depth below the sonobuoy from which analog signals were telemetered back to R/V CONRAD for digital recording at a sample interval of 1 millisecond. Each sonobuoy data trace recorded 9000 samples beginning at the shot instant. Only an anti-aliasing filter was applied to the data stream which was digitized and digitally recorded on the ship's data logger system along with the SCS, navigation, gravity and magnetics data..

Both the refraction and reflection experiments recorded the same seismic source consisting of two 80 cu. in. water gun fired at 12 second intervals for a nominal 50 m shot spacing at ~8 kts.

Individual sonobuoys were deployed along dip and strike lines throughout the ~3000 km of SCS profiling over the principal 120 x 230 km study area (Fig. 1). Buoys were recorded out to offset distances of ~10 km.

DATA ANALYSIS

The sonobuoy data and the coincident SCS data were stripped from the R/V CONRAD data logger tapes at LDGO and shipped to Austin in SEG-Y format. At UT Austin, the data were loaded on the Institute's VAX 11/780 running Cogniseis' DISCO™ seismic processing package. The data were reformatted, resampled at 2 ms intervals and edited to remove bad traces and to add missing traces. The two channel data (SCS and sonobuoy) were bandpass filtered (15 to 85 Hz) and plotted for inspection.

The SCS data are of excellent quality even considering the high ship speed (~8 kts) during acquisition. Signal penetration is remarkably good with arrivals down to 1.0 s subbottom visible in many areas. This resulted in spectacular dip line profiles of the Broken Ridge (e.g., Fig. 2) showing the thick northward dipping carbonate and chert sequence, truncated near the crest of the ridge by a horizontal angular unconformity. This unconformity is clearly overlain by a broad lens-shaped unit along the summit of the ridge. Pelagic sediments cap the ridge, draping down the northward slope where they lie conformably on the carbonate sequence. Drilling at DSDP Site 255, at the very crest of the ridge where the angular unconformity nearly outcrops, sampled limestones of Santonian age beneath a sequence of Middle Eocene detrital sands and gravels containing reworked Early Eocene materials. It is apparent from Line 20 of the SCS data that the limestones sampled lie well down from the inferred top of the carbonate and chert sequence and thus provide little control on the minimum age of the sequence or the onset of the rifting event which separated Broken Ridge from its conjugate, the Kerguelen Plateau.

The sonobuoy records show considerable subbottom structure although signal-to-noise ratios are low. Given the small size of the source and its high characteristic frequency, this is not surprising. Refracted crustal arrivals are largely invisible beyond 8 km offset in most cases (e.g., Fig. 3) however, deeper events visible in the SCS profile often produced visible precritical angle reflections in the sonobuoy records.

Twenty-two sonobuoys were selected for analysis in this study (Table 1) to meet the primary objective of providing velocity control in the vicinity of the proposed drill sites along line 20 (Fig. 2). Particular emphasis was placed on dip lines 10 and 20 and strike lines crossing these two, with additional coverage down dip (ie., to the north) from this area. This effectively sampled the lens of lagoonal sediments and the underlying truncated carbonate sequence at several latitudes and the northern slope of the plateau both proximal and distal to the lagoonal sedimentation environment. Figures 3-6 show buoy profiles representative of these various strikes.

Unfortunately, the low signal-to-noise ratios precluded the effective conversion of the travel time data to delay time data for automated inversion. Actual attempts yielded inconsequential bounds on solutions. Therefore, we resorted to forward modeling of the sonobuoy data using raytracing.

A computer program (Appendix I) was developed to simplify and to systematize the construction of the initial velocity model. The program uses a digitizing tablet upon which the sonobuoy plot is mounted for digitization of refracted and reflected events. The program automatically computes distance offsets and scales from the direct water wave phase, taking into account both the source and receiver depths. The water wave ranging is sensitive enough to distinguish the higher velocity of the shallow waters surrounding the receiver (~1.515 km/s) versus the lower water column sounding velocity (~1.490 km/s) required to model the seafloor reflection. The program makes use of both precritically reflected and/or refracted arrivals and aids in the objective identification and correlation of both. The SCS records provided valuable control on travel times for precritical reflection events as well as identifying areas of extreme lateral heterogeneity to be avoided.

After construction of the initial models, a raytracing program (Appendix II) was applied to each model, the results plotted for overlay, comparison and iterative model refinement. The SCS profiles allowed regional comparison of buoy models as a check on consistency. Results were sensitive to the depth difference between the SCS receiver (~10 m) and the sonobuoy receiver (122 m) suggesting the overall consistency of the technique.

Table 1.

RC27-08 Sonobuoys Analyzed in This Study.

Buoy	Line	CDP	Time	Date	Latitude	Longitude	Azim	Length
11	3	-8	17:29z	09 Sep 86	30° 57.28'S	93° 28.21'E	105°	12.6km
12	3	347	18:40	09 Sep 86	31° 00.49'	93° 42.88'	104°	12.0
15	4	1612	07:52	10 Sep 86	30° 59.21'	94° 03.44'	284°	8.2
16	4	1847	08:39	10 Sep 86	30° 57.15'	93° 53.38'	283°	10.2
17	4	2195	09:49	10 Sep 86	30° 54.00'	93v 38.21'	281°	7.9
32	10	2108	01:18	14 Sep 86	30° 54.91'	93° 47.78'	196°	5.4
33	10	2308	01:58	14 Sep 86	31° 00.34'	93° 45.76'	197°	5.2
34	12	771	04:25	15 Sep 86	30° 32.96'	93° 48.95'	282°	9.9
35	12	1375	06:26	15 Sep 86	30° 28.42'	93° 28.33'	279°	7.8
36	12	1642	07:19	15 Sep 86	30° 27.01'	93° 18.09'	278°	8.6
38	13	1160	12:32	15 Sep 86	30° 24.82'	93° 55.38'	101°	9.2
39	13	1461	13:33	15 Sep 86	30° 26.99'	94° 06.79'	102°	10.0
47	16	254	07:56	16 Sep 86	30° 10.15'	94° 01.47'	194°	11.4
48	16	722	09:30	16 Sep 86	30° 22.16'	93° 56.85'	188°	9.7
49	16	1345	11:35	16 Sep 86	30° 38.89'	93° 53.69'	194°	7.9
50	17	39	13:23	16 Sep 86	30° 52.01'	93° 47.05'	279°	8.8
51	17	367	14:29	16 Sep 86	30° 49.61'	93° 34.80'	281°	8.7
54	18	2232	06:47	17 Sep 86	30° 50.46'	94° 19.17'	104°	9.7
55	18	3592	11:20	17 Sep 86	30° 54.94'	94° 56.57'	010°	10.5
59	20	739	21:38	17 Sep 86	30° 48.12'	93° 36.98'	193°	6.0
60	20	957	22:22	17 Sep 86	30° 54.45'	93° 34.23'	190°	5.2
61	20	1154	23:02	17 Sep 86	31° 00.25'	93° 32.78'	189°	5.7

RESULTS AND CONCLUSIONS

The sonobuoy data delineate precritical reflections and wide angle arrivals from the Neogene-to-Cretaceous sedimentary sequence. Total sub-seafloor penetration approaches 2.5 km although arrivals from the rough acoustic (presumably basaltic) basement are generally very weak if present. There is very good correlation of most of the precritical reflection events with the mappable horizons observed in the SCS data. No low velocity zones were observed anywhere in the data.

The modeling results for the 22 sonobuoys are presented in figures 7a-7v. These results can be summarized in terms of three average models which characterize the general structure of the Broken Ridge. The velocities and reflection character of the significant layers are given in Table 2. All of the models share a common suite of velocities in their lower sections, beginning with 3.1 km/s. The principal differences between the models lie in the velocities of the the shallower units and the subseafloor depth to the 3.1 km/s layer.

Table 2

Summary of Earth Model Results

Model A (e.g., buoy 54)		Model B (e.g., buoy 17)		Model C (e.g., buoy 12)	
Velocity	Unit	Velocity	Unit	Velocity	Unit
1.49	water	1.49	water	1.49	water
1.60 ±0.10	pelagics	1.60 ±0.10	pelagics	1.60 ±0.10	pelagics
1.85 ±0.05	downslope laminated unit	1.65 ±0.05	lagoonal unit	1.65 ±0.05	lagoonal unit
2.10 ±0.10	finely laminated unit	2.10 ±0.10	finely laminated unit		
3.10 ±0.10	coarsely laminated unit	3.10 ±0.10	coarsely laminated unit	3.10 ±0.10	coarsely laminated unit
3.60 ±0.10	base of laminated unit	3.60 ±0.10	base of laminated unit	3.60 ±0.10	base of laminated unit
4.40 ±0.2	deep discontinuous event	4.40 ±0.2	deep discontinuous event	4.40 ±0.2	deep discontinuous event
4.80 ±0.15	deep event (crest outcrop)	4.80 ±0.15	deep event (crest outcrop)	4.80 ±0.15	deep event (crest outcrop)
>5.3	deepest event (very weak)	>5.3	deepest event (very weak)	>5.3	deepest event (very weak)

The first, model A, describes the stratigraphy of the plateau's northern slope, away from the angular unconformity. Sonobuoy 54 is representative of this class of model. The thin pelagic sequence overlies a thickening (to the North) wedge of presumably mid-Eocene sediments shed by the eroding summit of Broken Ridge as it is uplifted by the rifting process. This sequence has a relatively low velocity of 1.85 km/s. Beneath this unit lies a transparent wedge (thinning to the North) with a similar velocity of 1.8-1.9 km/s. However, this unit is truncated by the angular unconformity; thus it is most likely an early rift or pre-rift facies. This unit is conformal to the underlying finely laminated carbonate sequence which yields a velocity of 2.1 km/s. This layer is relatively uniform in thickness, perhaps thinning slowly to the North. The relatively uniform velocity structure of the dipping carbonate section begins below this layer.

Model B is representative of the northern edge of the plateau scarp over the angular unconformity (e.g., sonobuoy 17). Here, the flat lying lagoonal sequence (velocity 1.65 km/s) underlies the pelagic cover. Lying unconformably on the finely laminated unit, the velocity increases at its base to 2.1 km/s. The shoaling of the finely laminated unit is quite apparent in a visual comparison of the refraction profiles for sonobuoys 54, 50 and 17 (Figs. 3-5). Otherwise, this model resembles model A in its lower sections.

Model C samples the southern portion of the angular unconformity where the 2.2 km/s unit has been fully removed. Thus velocities climb rapidly to 3. 1 km/s. Drill site BR-4 is located to the South of this position and should sample even higher velocity rocks (4.4 km/s?) at the angular unconformity. The deep event, 4.8 km/s appears to correlates well with the outcrop which bounds the southern margin of the lagoonal sequence.

In conclusion, the velocities in the carbonate and chert sequence along the unconformity are observed to increase to the south as stratigraphically deeper sections of the carbonate are exposed. This suggests that the higher velocities may represent increased diagenesis rather than compaction and dewatering. The seismic structure derived here indicate that the planned series of nominally 400 m deep holes should be able to sample a significant portion of the dipping carbonate and chert sequence along the angular unconformity.

FIGURE CAPTIONS

Figure 1. Track chart of R/V CONRAD Cruise RC2708 over the Broken Ridge in the Eastern Indian Ocean. The thin line denotes the ship's track annotated with hourly ('+') and daily ('x') tic marks. The heavy lines indicate the seismic refraction line locations while the solid dots indicate the sonobuoy deployment position

Figure 2. The single channel seismic reflection section recorded along line 20 showing a north-to-south dip line profile of Broken Ridge. This seismic section (as well as those in Figs. 3-6) has been bandpass filtered (15-88 Hz) and plotted with a 500 ms AGC window. Travel times are in seconds. This 54-km-long profile clearly demarcates the the tilt-block structure of this oceanic plateau. The steeply dipping carbonate sequence is truncated by a flat-lying lens of "lagoonal" sediments at the crest of the ridge, forming a spectacular angular unconformity. Crossings with strike lines 3, 4, 17 and 18 are indicated as are the original four proposed ODP Leg 121 drill sites, BR-1 thru BR-4.

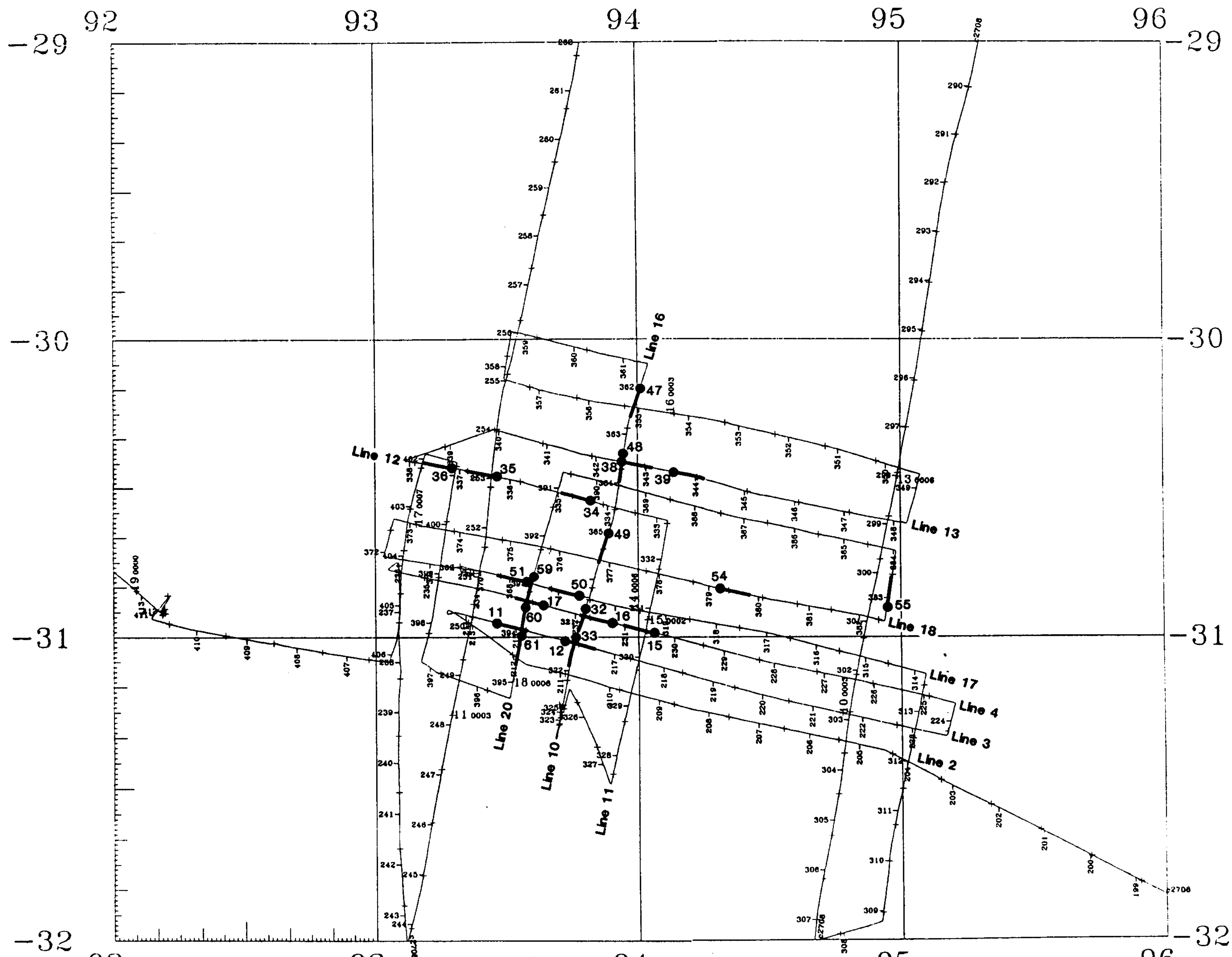
Figure 3. The seismic refraction travel time versus shot point (FFID) plot for sonobuoy 54. This buoy was deployed along strike on line 18. The small arrow denotes zero distance offset (i.e., the point of buoy deployment) while the heavy bar represents 5 km of offset. Travel times are in seconds. This refraction line samples the "normal" slope of Broken Ridge, well to the north of the angular unconformity at the plateau's crest (see Fig. 2). The relatively straight arrival is the direct water wave. The strong arrival approximately 0.16 s two-way travel time subbottom is the free surface ("ghost") reflection echo of the seafloor arrival recorded by the 122-m-deep hydrophone

Figure 4. The seismic refraction travel time versus shot point (FFID) plot for sonobuoy 50. This buoy was deployed along strike on line 17. The small arrow denotes zero distance offset, while the heavy bar represents 5 km of offset. Travel times are in seconds. This refraction line samples the northern edge of the "lagoonal" sediment lens overlying the angular unconformity at the plateau's crest (see Fig. 2).

Figure 5. The seismic refraction travel time versus shot point (FFID) plot for sonobuoy 17. This buoy was deployed along strike on line 4. The small arrow denotes zero distance offset while the heavy bar represents 5 km of offset. Travel times are in seconds. This refraction line samples the middle of the "lagoonal" sediment lens overlying the angular unconformity at the plateau's crest (see Fig. 2).

Figure 6. The seismic refraction travel time versus shot point (FFID) plot for sonobuoy 12. This buoy was deployed along strike on line 3. The small arrow denotes zero distance offset, while the heavy bar represents 5 km of offset. Travel times are in seconds. This refraction line samples the southern edge of the "lagoonal" sediment lens overlying the angular unconformity at the plateau's crest (see Fig. 2).

Figure 7. Plots of sonobuoy two-way travel time versus offset for each of the 22 sonobuoys analyzed in this study and listed in Table 1. Both precritical reflections and refractions are plotted. The former are annotated with the layer velocity in km/s. The inset contains a listing and plot of the final model as velocity versus two-way travel time (relative to the SCS receiver). The X-axis scale varies from buoy to buoy, reflecting a constant number of traces (shots) per inch but a varying ship speed (shot interval). Note that both the line number and the CDP (=FFID) at which the sonobuoy was deployed are given. The plots are reduced to 73% of their original scale. Enlargement to 137% of current size should produce plots at the original scale (0.2 s/in, 10 traces/in).



92
93
94
95
96
−29
−30
−31
−32
Line 2
Line 3
Line 4
Line 10
Line 11
Line 12
Line 13
Line 16
Line 17
Line 18
Line 20

Fig. 1

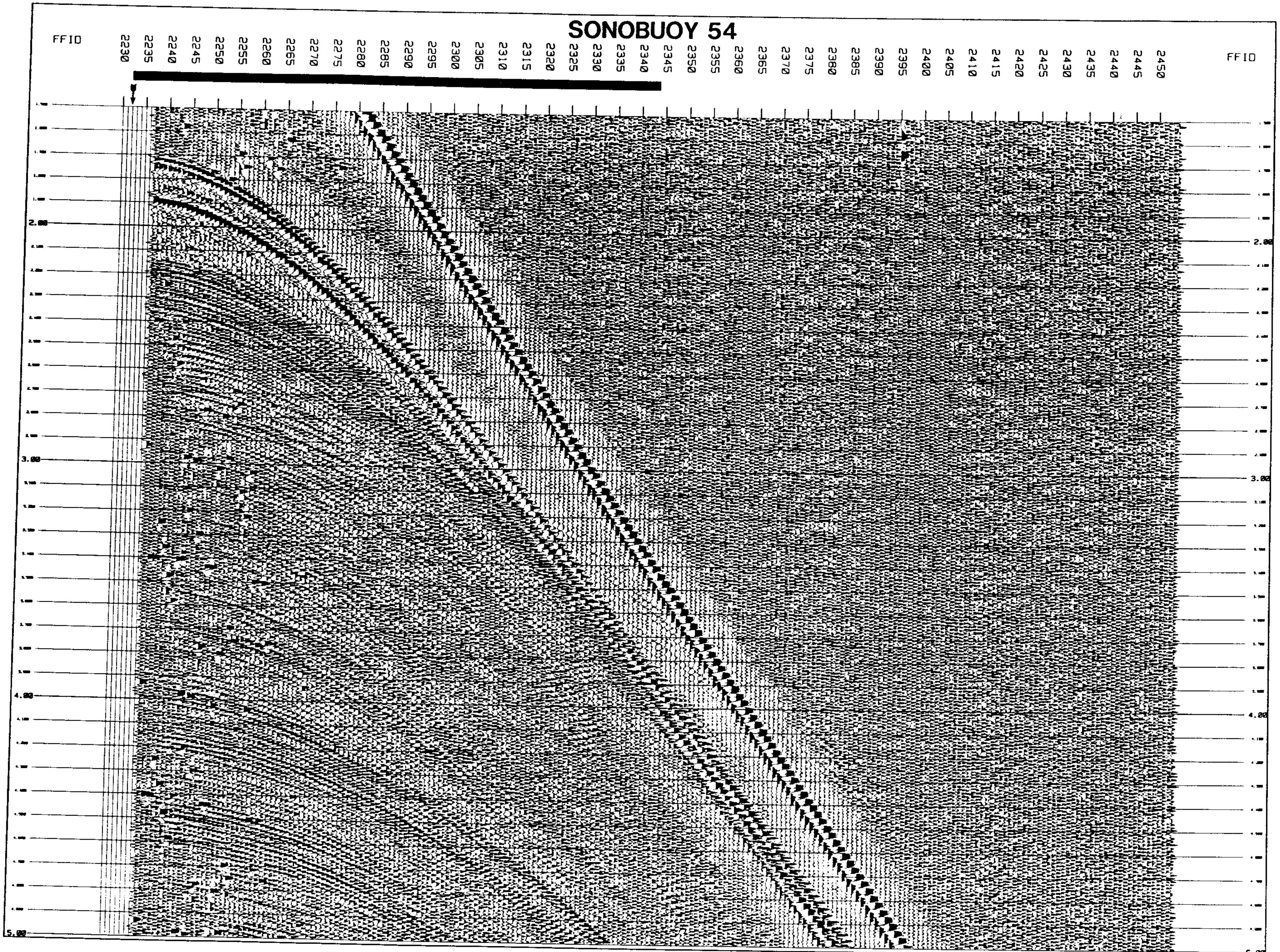

SONOBUOY 54
FFID
2230 2235 2240 2245 2250 2255 2260 2265 2270 2275 2280 2285 2290 2295 2300 2305 2310 2315 2320 2325 2330 2335 2340 2345 2350 2355 2360 2365 2370 2375 2380 2385 2390 2395 2400 2405 2410 2415 2420 2425 2430 2435 2440 2445 2450
2.00
3.00
4.00
5.00

Fig. 3

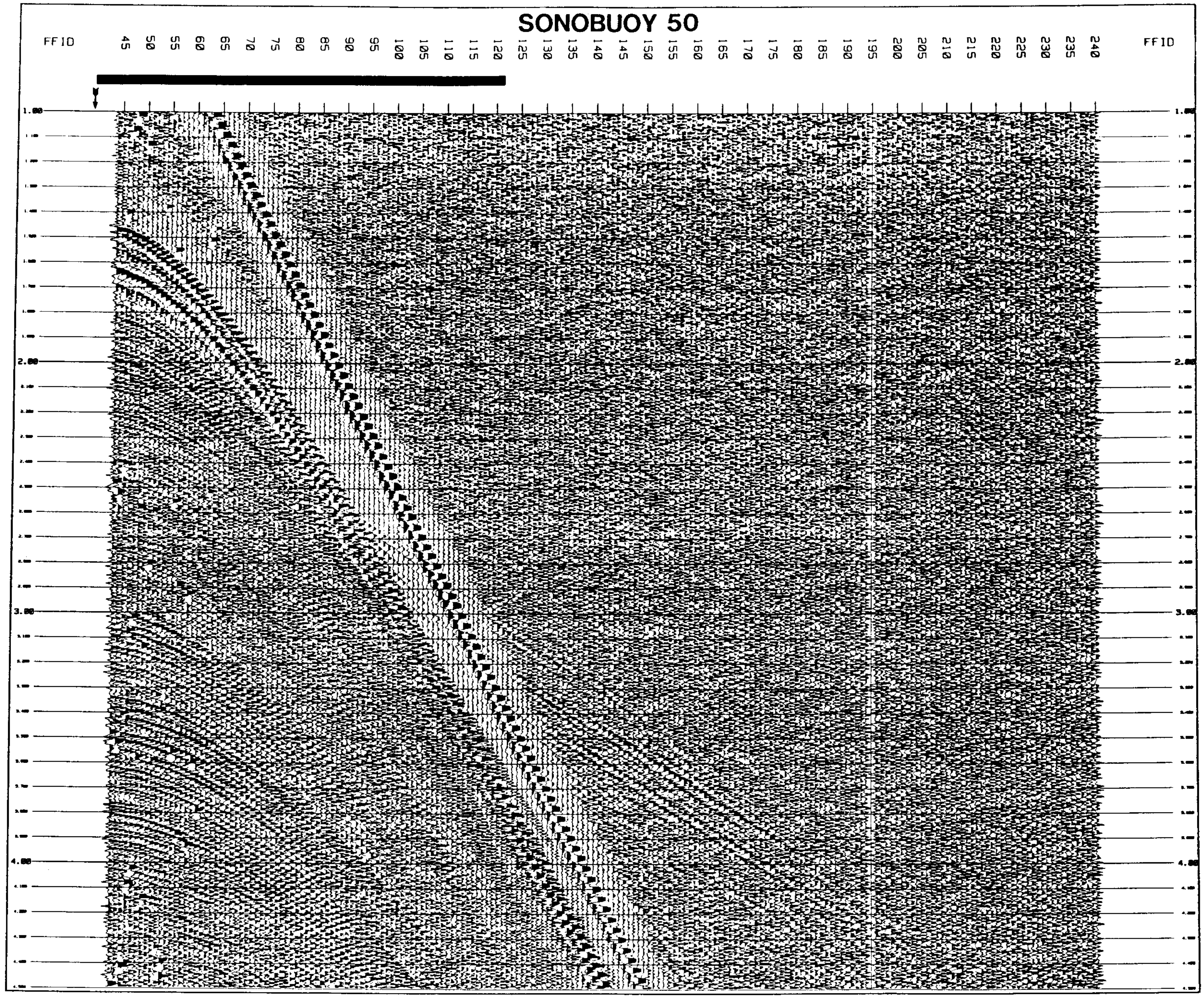

SONOBUOY 50
FFID
45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150 155 160 165 170 175 180 185 190 195 200 205 210 215 220 225 230 235 240
1.00
2.00
3.00
4.00

Fig.4

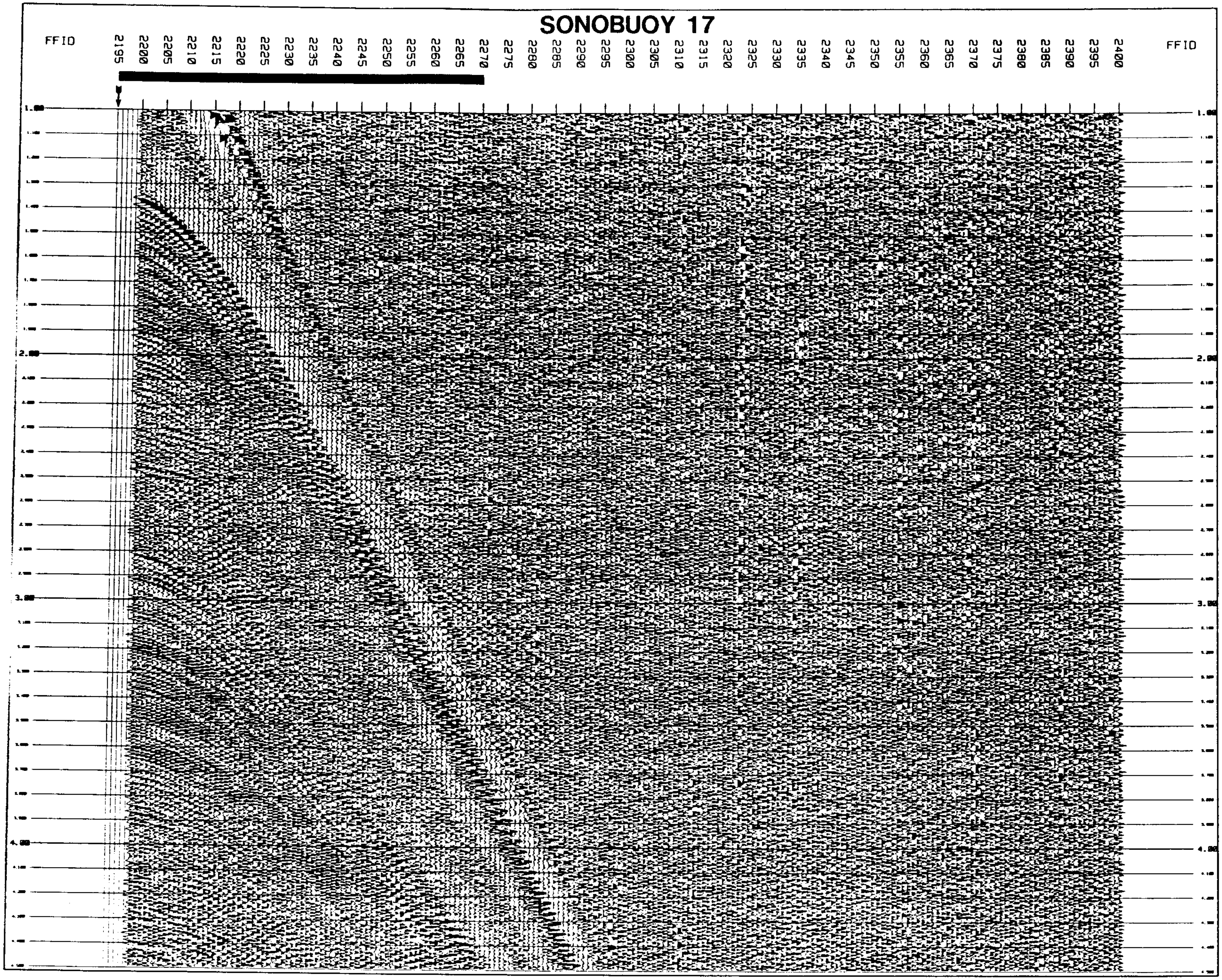

SONOBUOY 17
FFID
2195
2200
2205
2210
2215
2220
2225
2230
2235
2240
2245
2250
2255
2260
2265
2270
2275
2280
2285
2290
2295
2300
2305
2310
2315
2320
2325
2330
2335
2340
2345
2350
2355
2360
2365
2370
2375
2380
2385
2390
2395
2400
1.00
2.00
3.00
4.00

Fig. 5

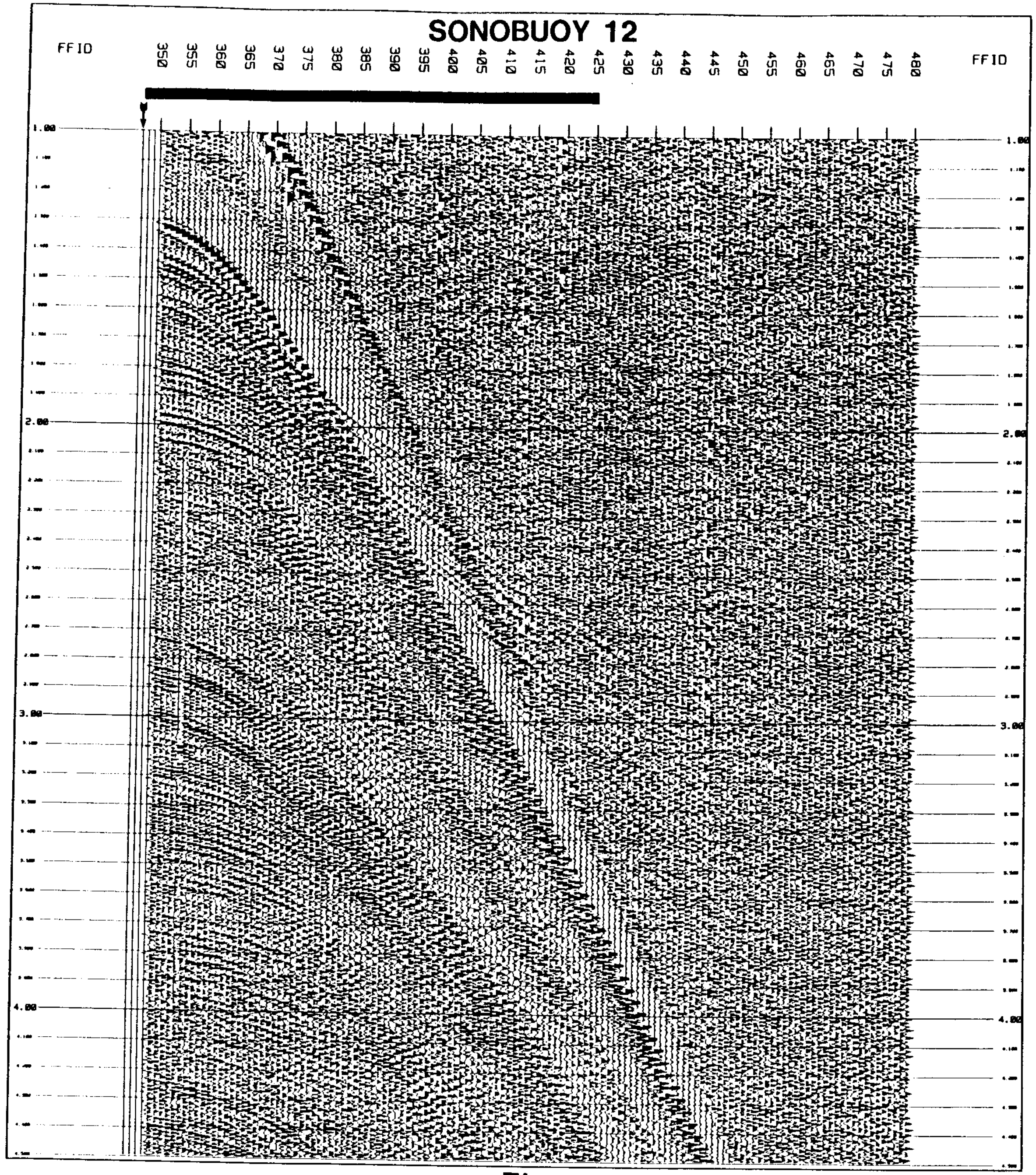

SONOBUOY 12
FFID
350 355 360 365 370 375 380 385 390 395 400 405 410 415 420 425 430 435 440 445 450 455 460 465 470 475 480
1.00
2.00
3.00
4.00

Fig. 6

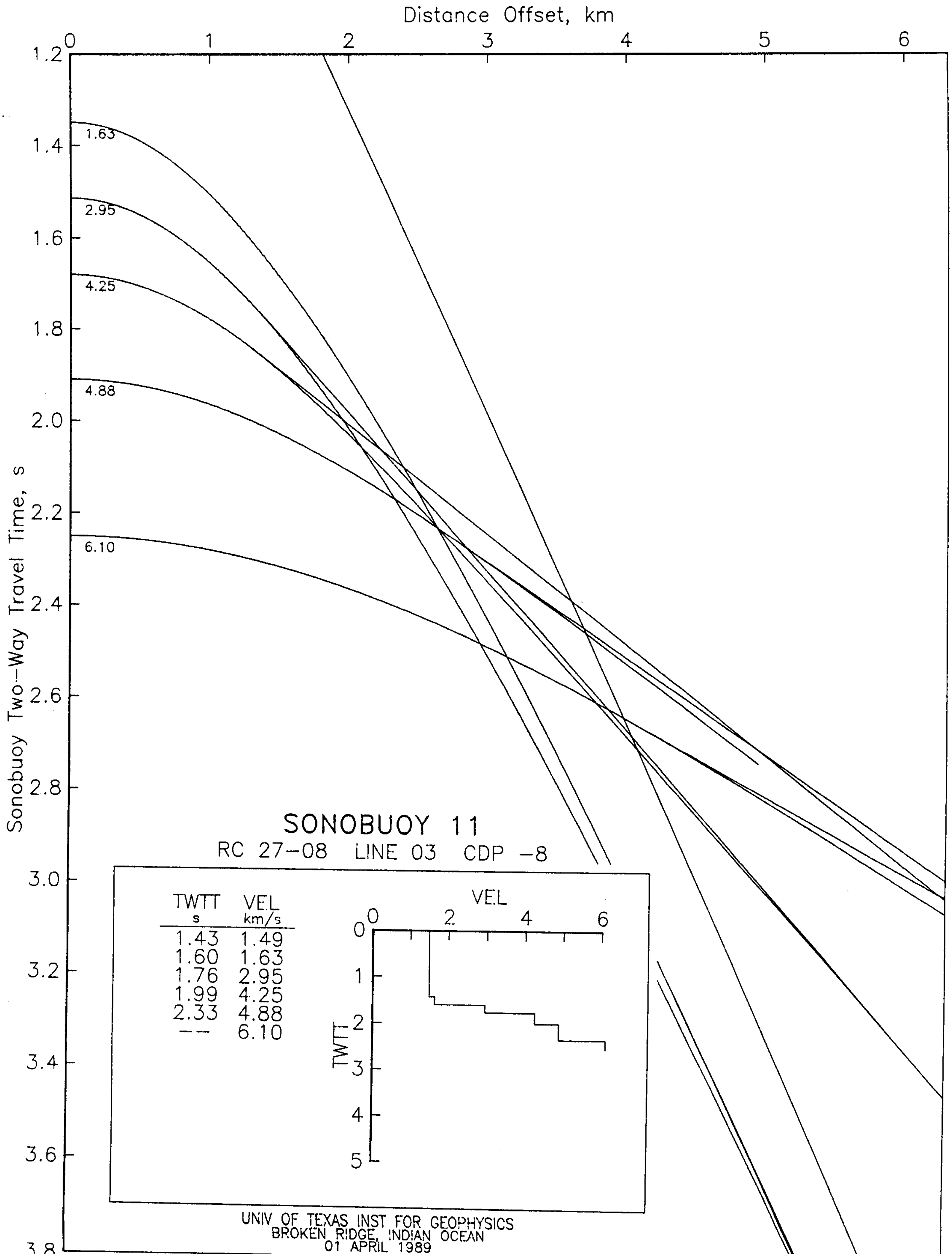
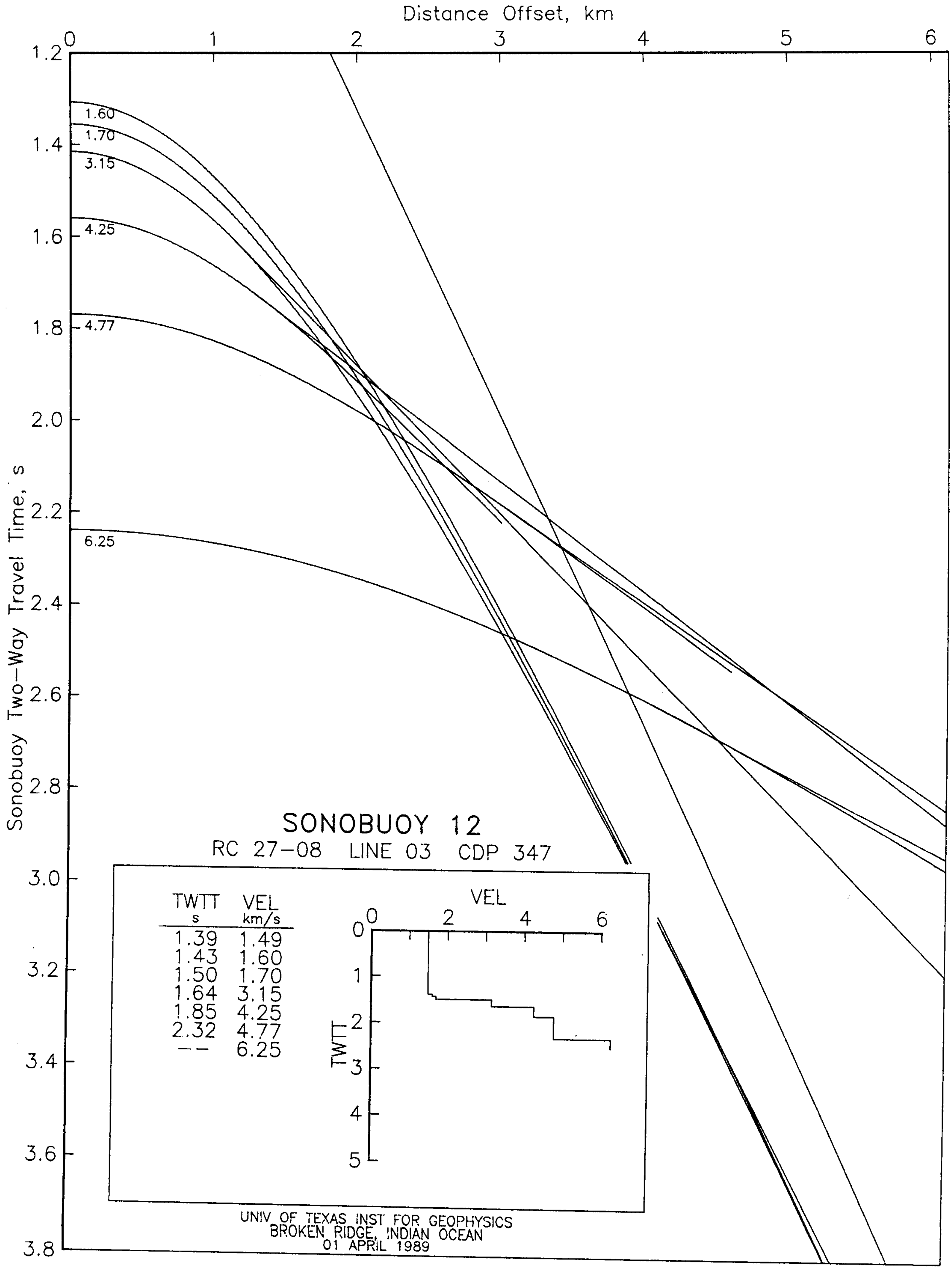
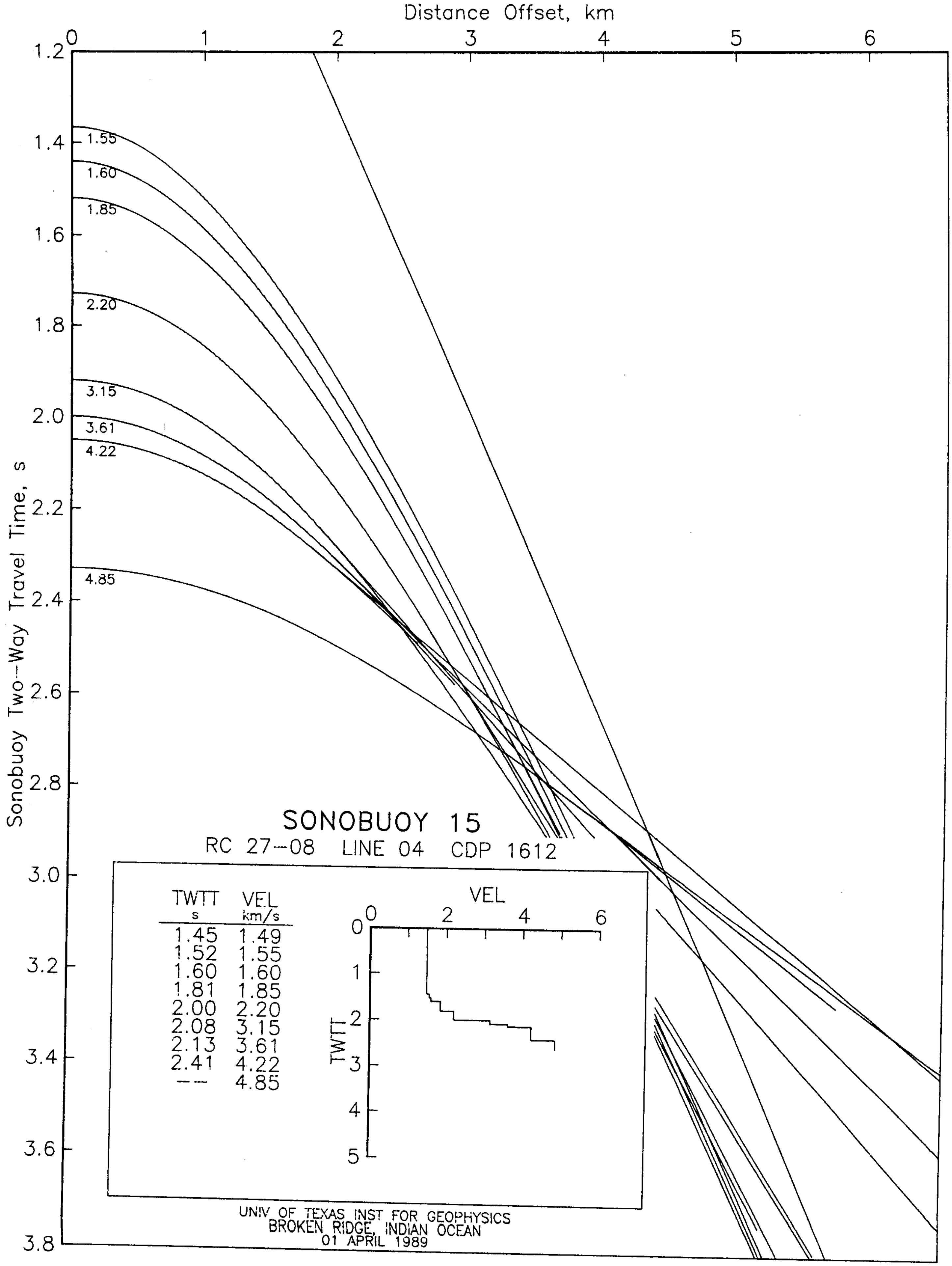
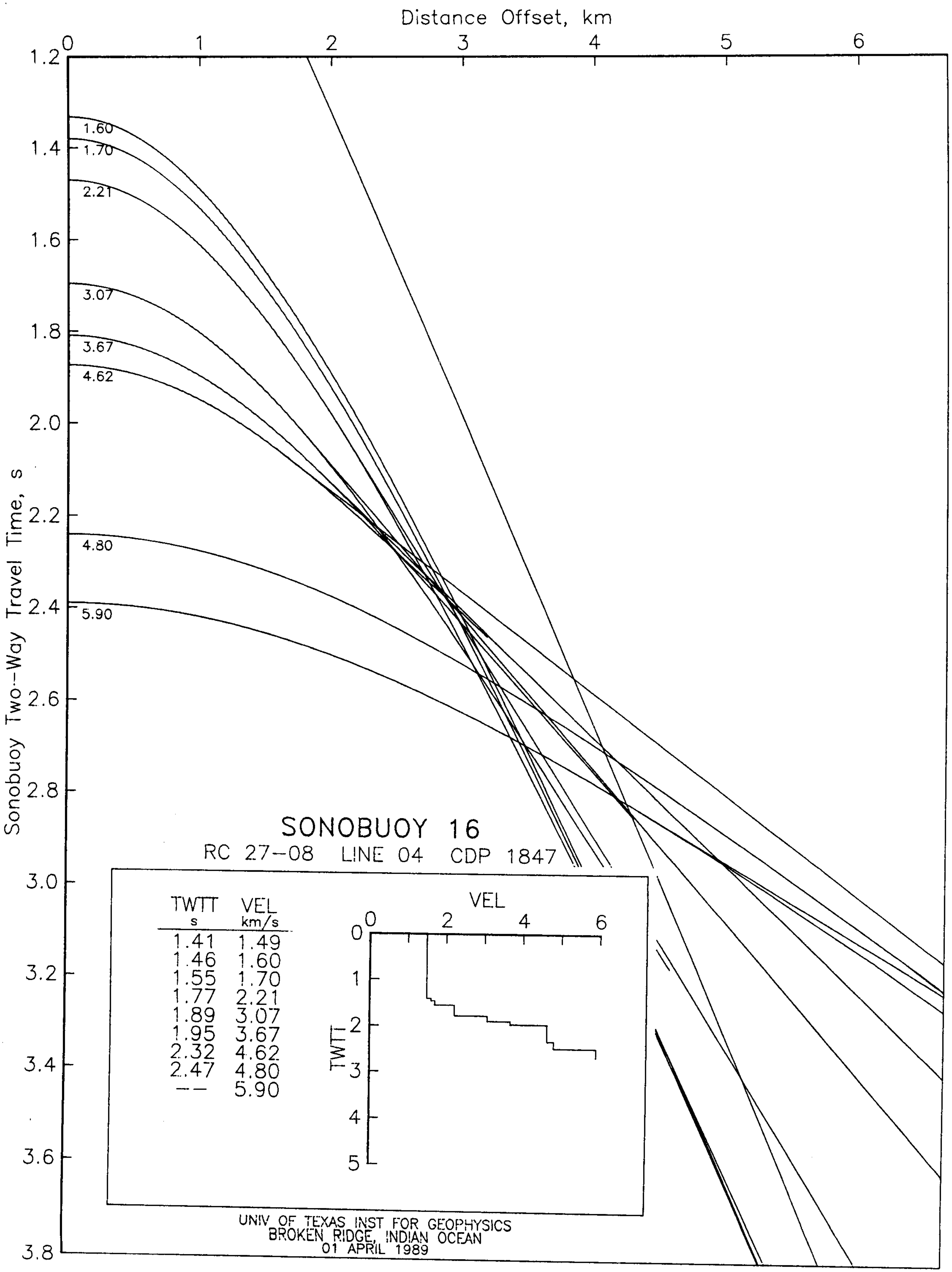
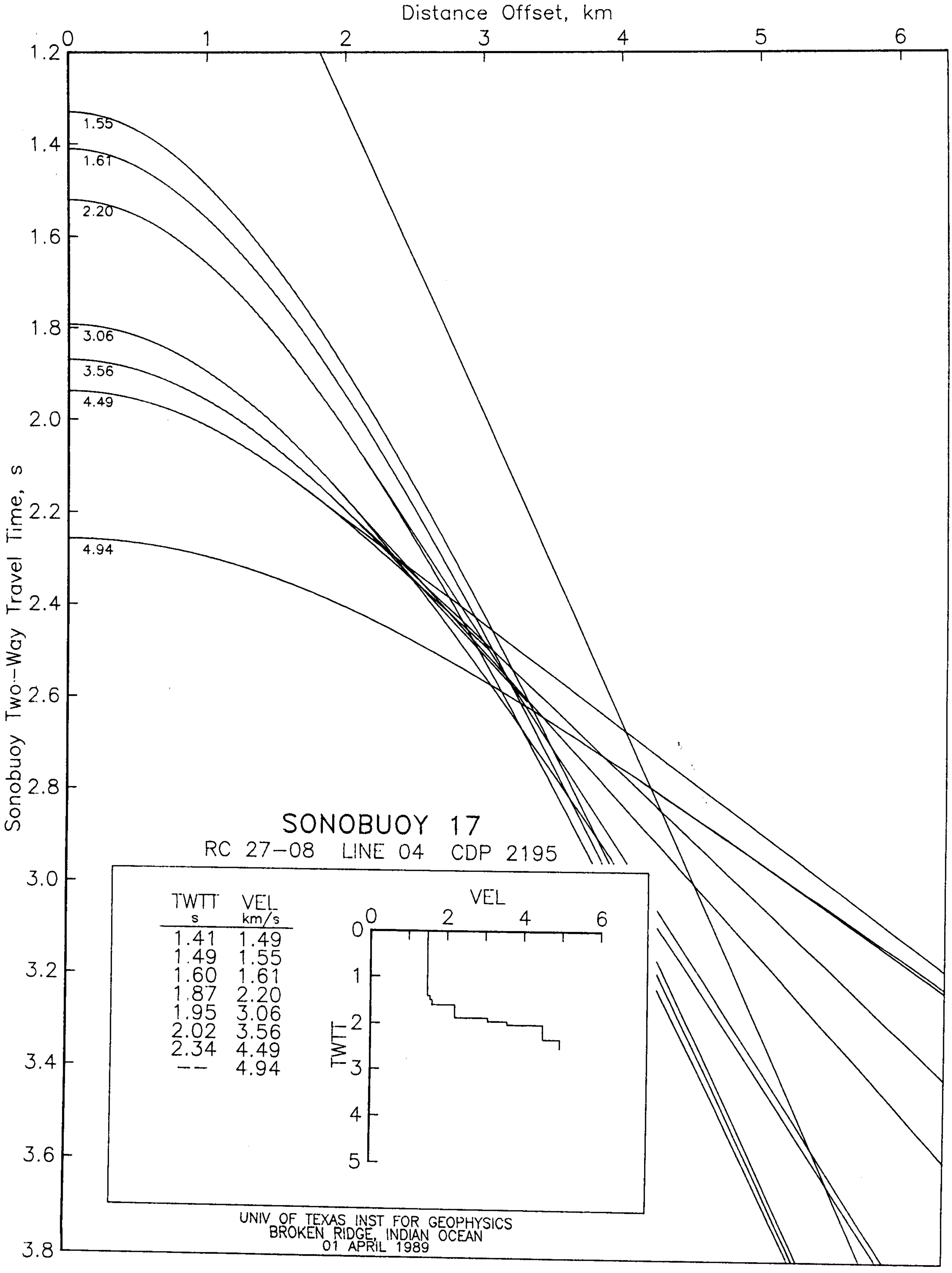

Distance Offset, km
0
1
2
3
4
5
6
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
Sonobuoy Two-Way Travel Time, s
1.63
2.95
4.25
4.88
6.10
SONOBUOY 11
RC 27-08 LINE 03 CDP -8
TWTT s
VEL km/s
1.43 1.49
1.60 1.63
1.76 2.95
1.99 4.25
2.33 4.88
-- 6.10
VEL
0
2
4
6
0
1
2
3
4
5
TWTT
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

Fig. 7 A–V

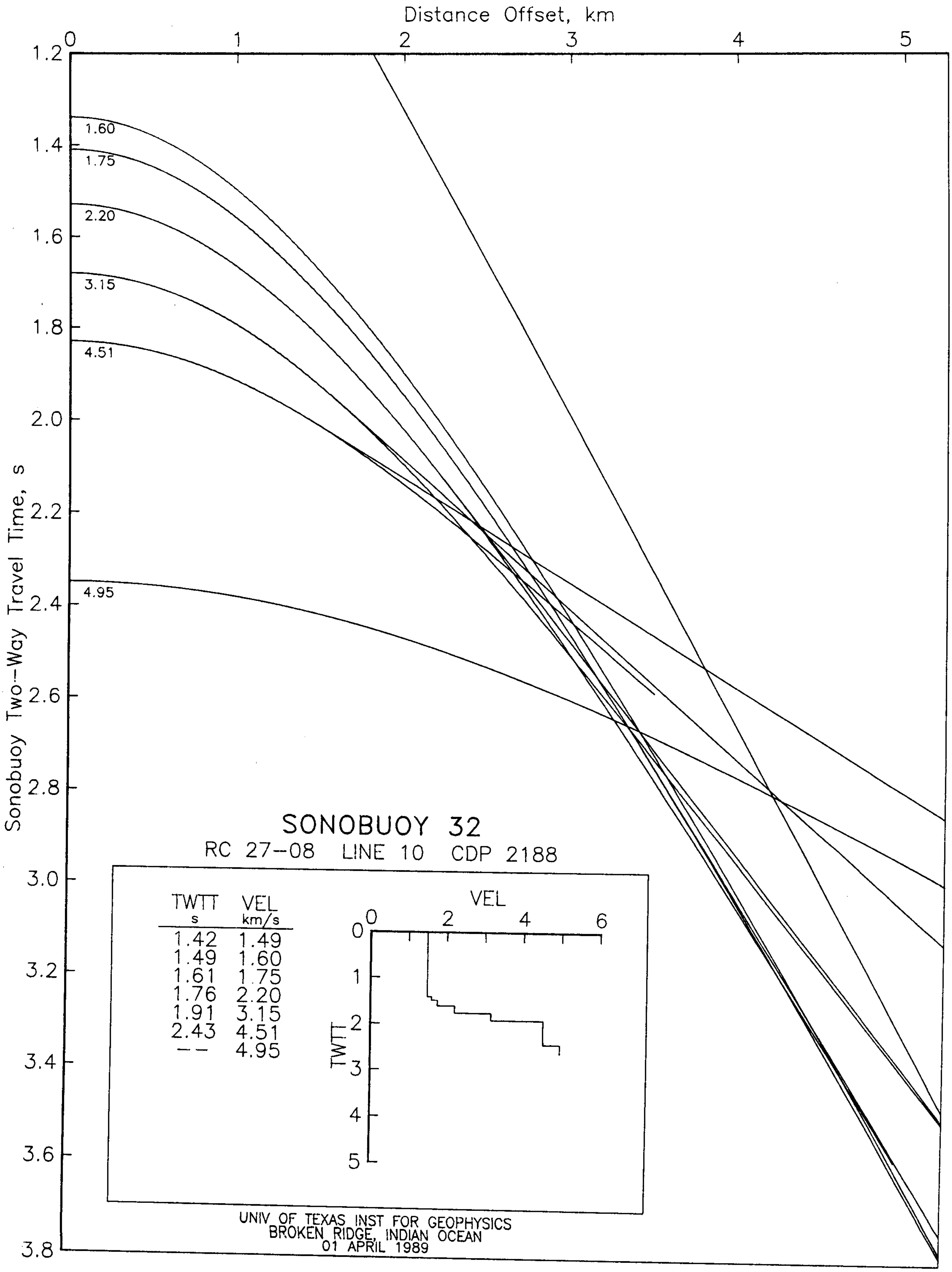

Distance Offset, km
0
1
2
3
4
5
6
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
Sonobuoy Two-Way Travel Time, s
1.60
1.70
3.15
4.25
4.77
6.25
SONOBUOY 12
RC 27-08 LINE 03 CDP 347
TWTT s
VEL km/s
1.39 1.49
1.43 1.60
1.50 1.70
1.64 3.15
1.85 4.25
2.32 4.77
-- 6.25
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

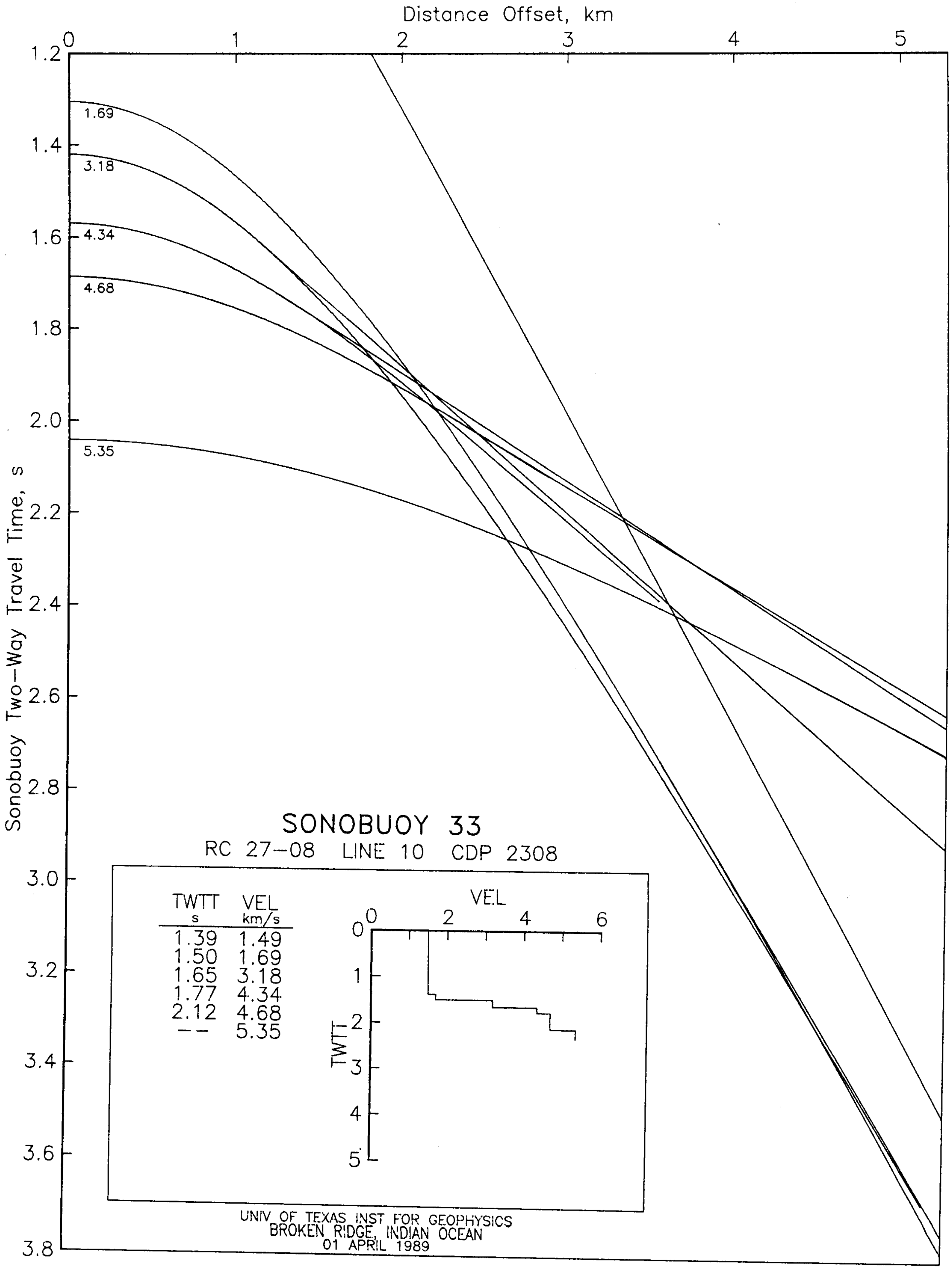

Distance Offset, km
0
1
2
3
4
5
6
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
Sonobuoy Two-Way Travel Time, s
1.55
1.60
1.85
2.20
3.15
3.61
4.22
4.85
SONOBUOY 15
RC 27-08 LINE 04 CDP 1612
TWTT s
VEL km/s
1.45 1.49
1.52 1.55
1.60 1.60
1.81 1.85
2.00 2.20
2.08 3.15
2.13 3.61
2.41 4.22
-- 4.85
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
0 1 2 3 4 5 6
Sonobuoy Two-Way Travel Time, s
1.2 1.4 1.6 1.8 2.0 2.2 2.4 2.6 2.8 3.0 3.2 3.4 3.6 3.8
1.60
1.70
2.21
3.07
3.67
4.62
4.80
5.90
SONOBUOY 16
RC 27-08 LINE 04 CDP 1847
TWTT s
VEL km/s
1.41 1.49
1.46 1.60
1.55 1.70
1.77 2.21
1.89 3.07
1.95 3.67
2.32 4.62
2.47 4.80
-- 5.90
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

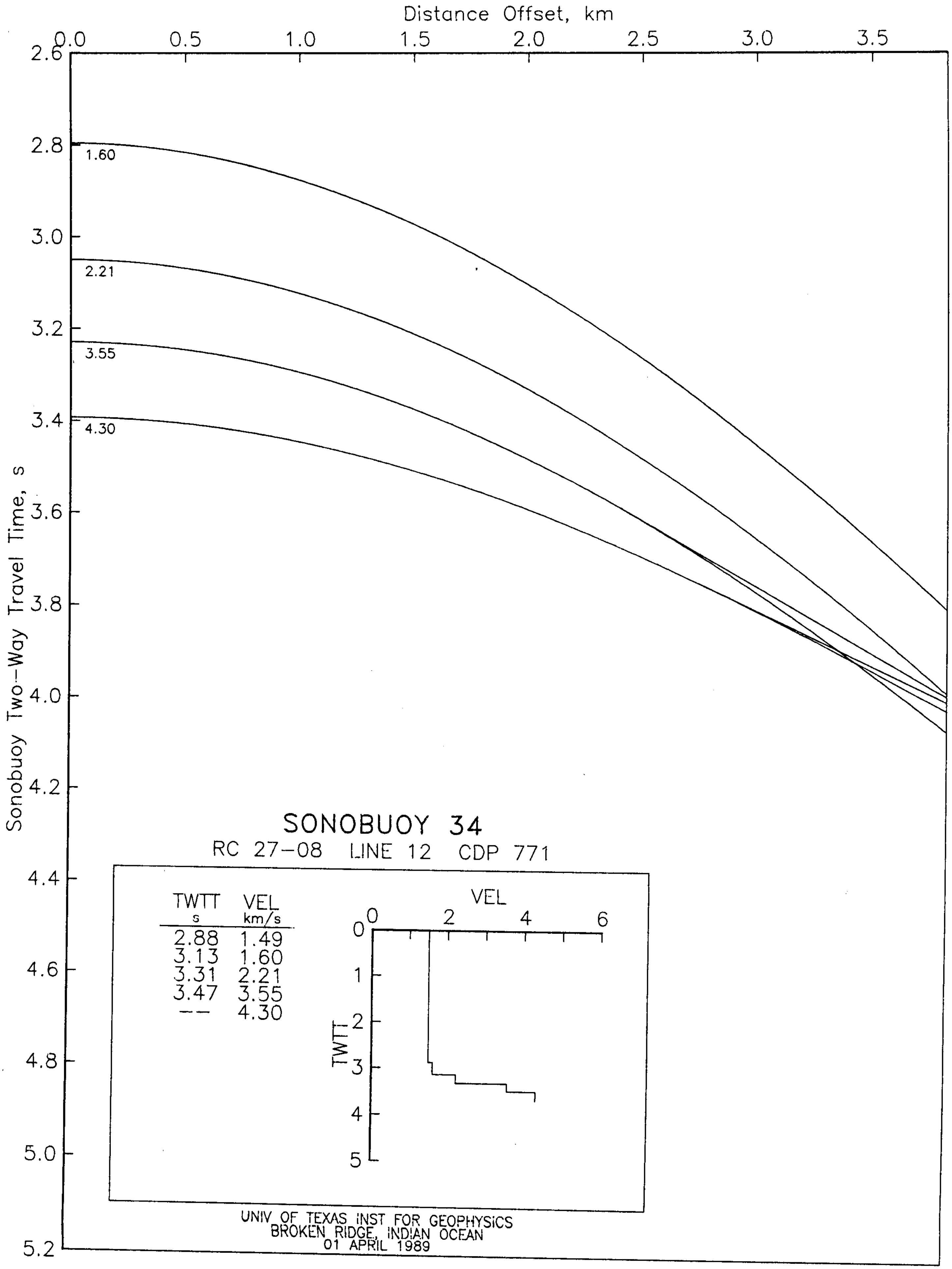


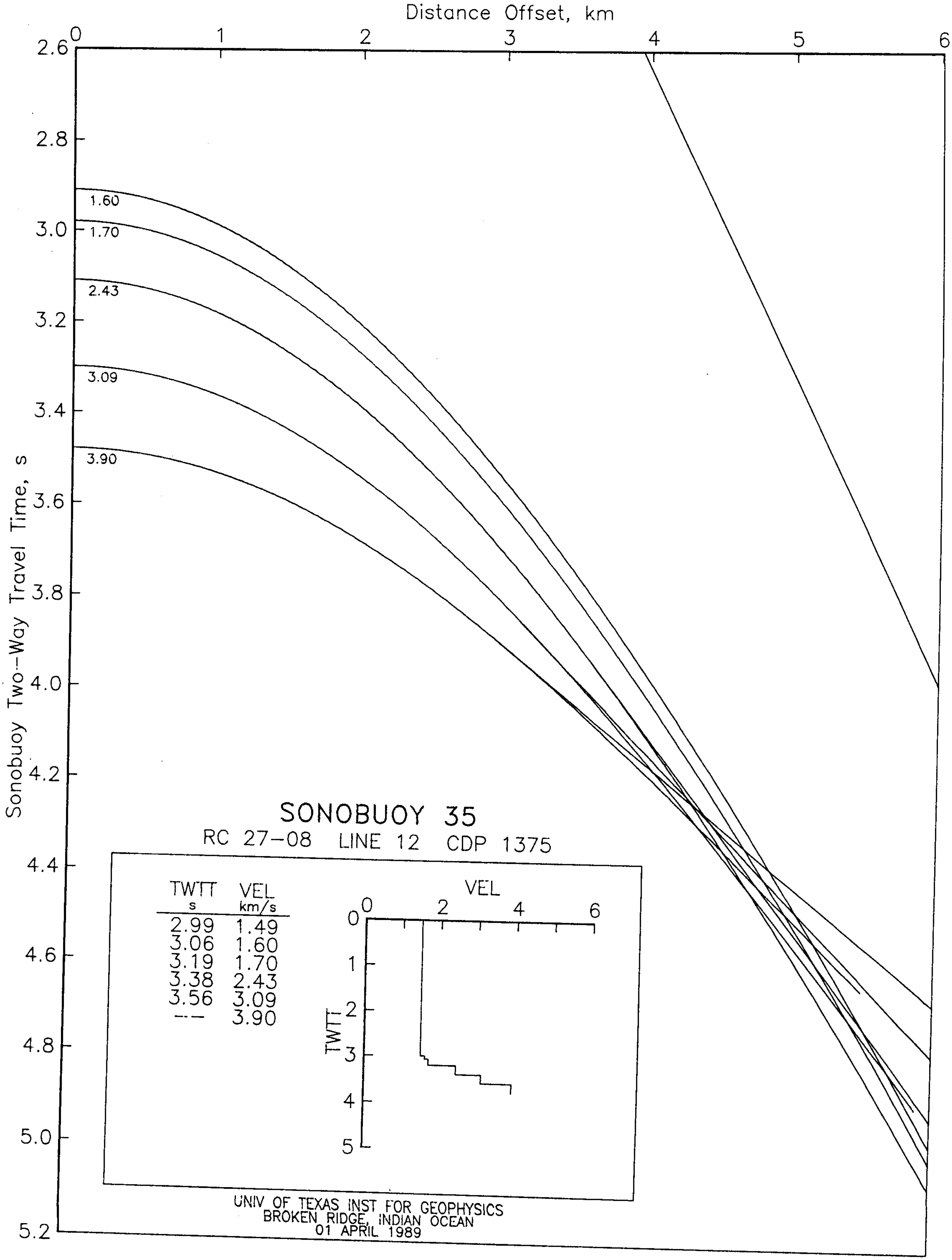
Distance Offset, km
0 1 2 3 4 5 6
Sonobuoy Two-Way Travel Time, s
1.2 1.4 1.6 1.8 2.0 2.2 2.4 2.6 2.8 3.0 3.2 3.4 3.6 3.8
1.55
1.61
2.20
3.06
3.56
4.49
4.94
SONOBUOY 17
RC 27-08 LINE 04 CDP 2195
TWTT s
VEL km/s
1.41 1.49
1.49 1.55
1.60 1.61
1.87 2.20
1.95 3.06
2.02 3.56
2.34 4.49
-- 4.94
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
1.60
1.75
2.20
3.15
4.51
4.95
SONOBUOY 32
RC 27–08 LINE 10 CDP 2188
TWTT s
VEL km/s
1.42 1.49
1.49 1.60
1.61 1.75
1.76 2.20
1.91 3.15
2.43 4.51
– – 4.95
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

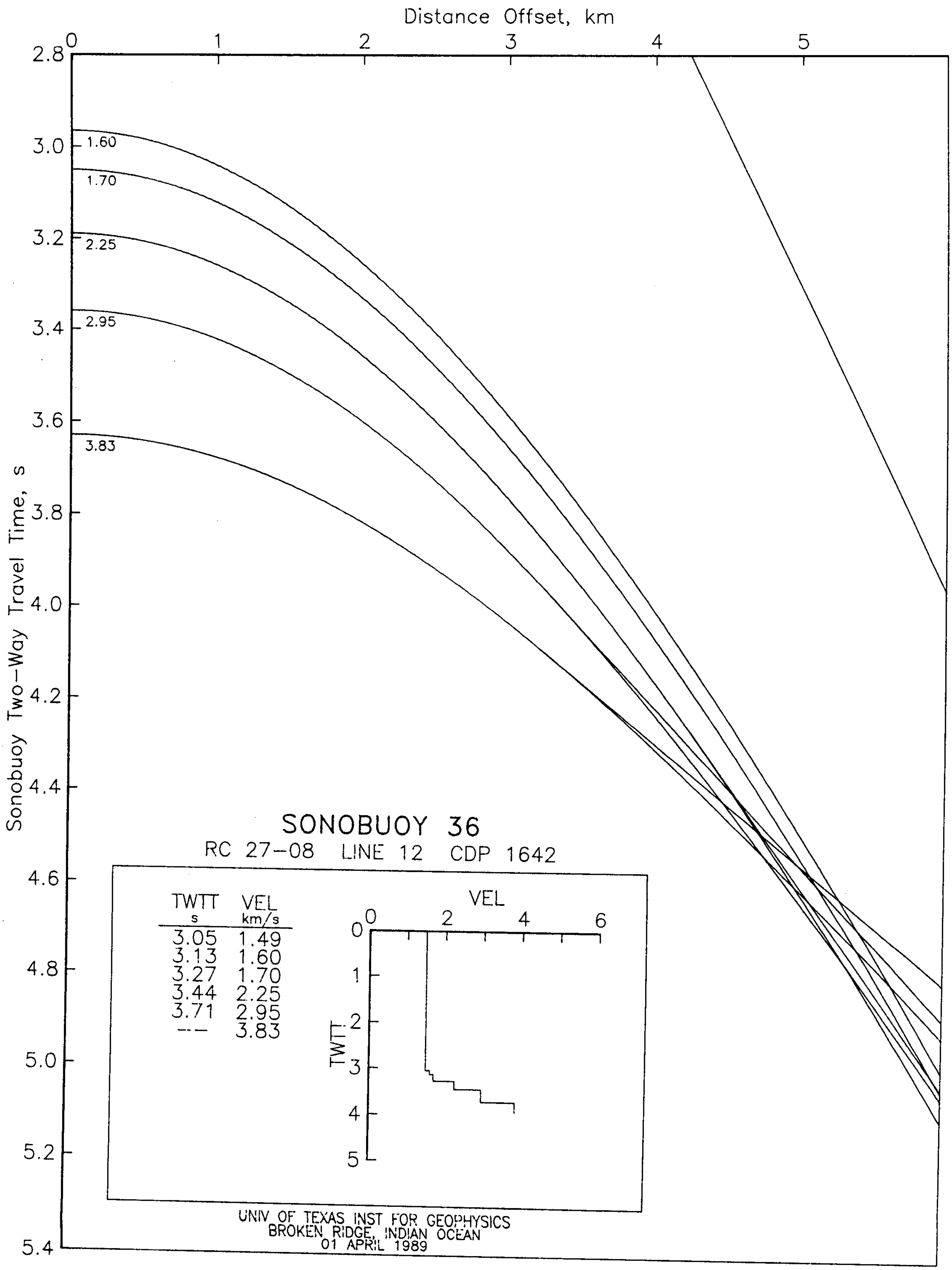


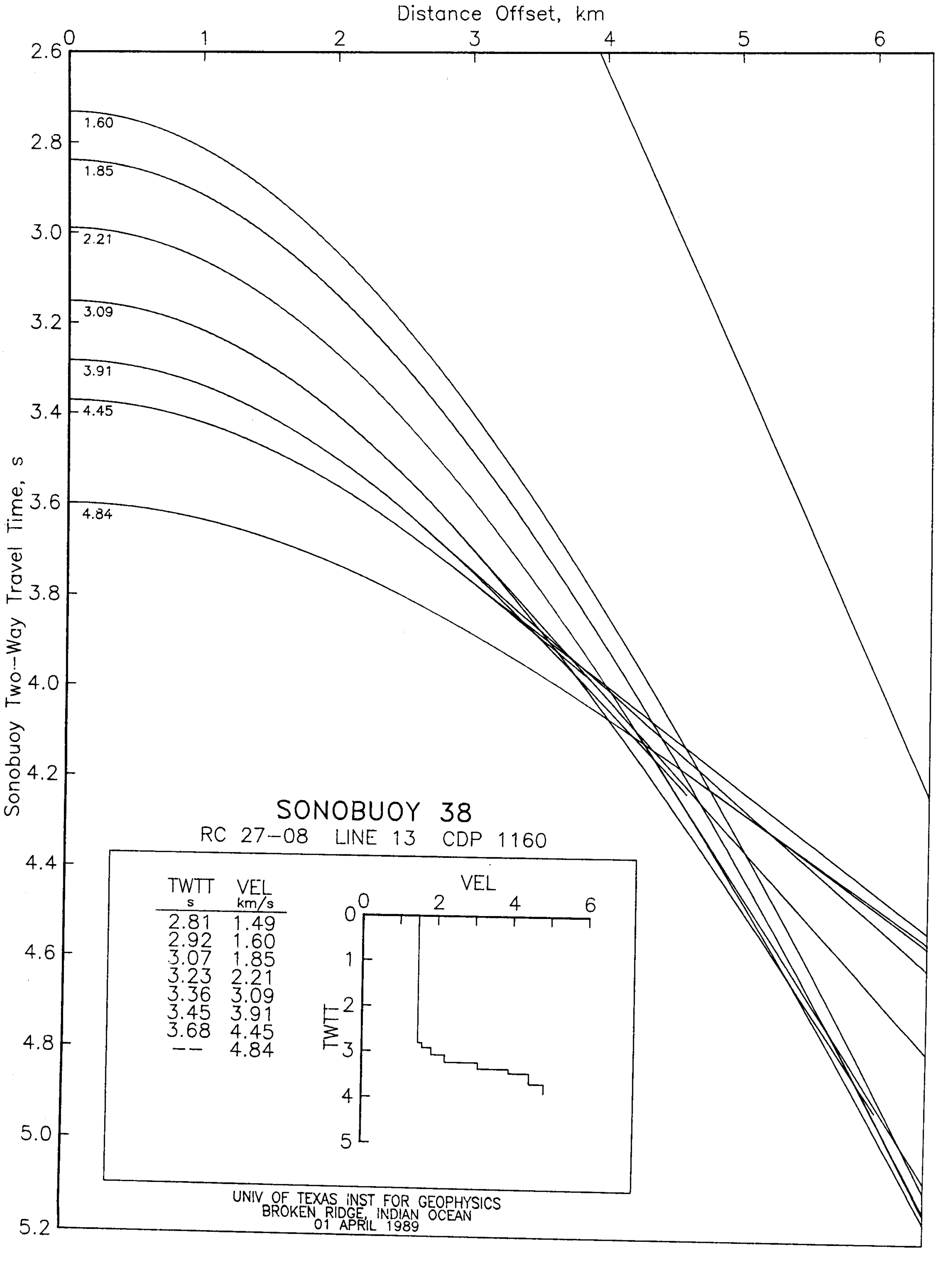
Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
1.69
3.18
4.34
4.68
5.35
SONOBUOY 33
RC 27-08 LINE 10 CDP 2308
TWTT s
VEL km/s
1.39 1.49
1.50 1.69
1.65 3.18
1.77 4.34
2.12 4.68
-- 5.35
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

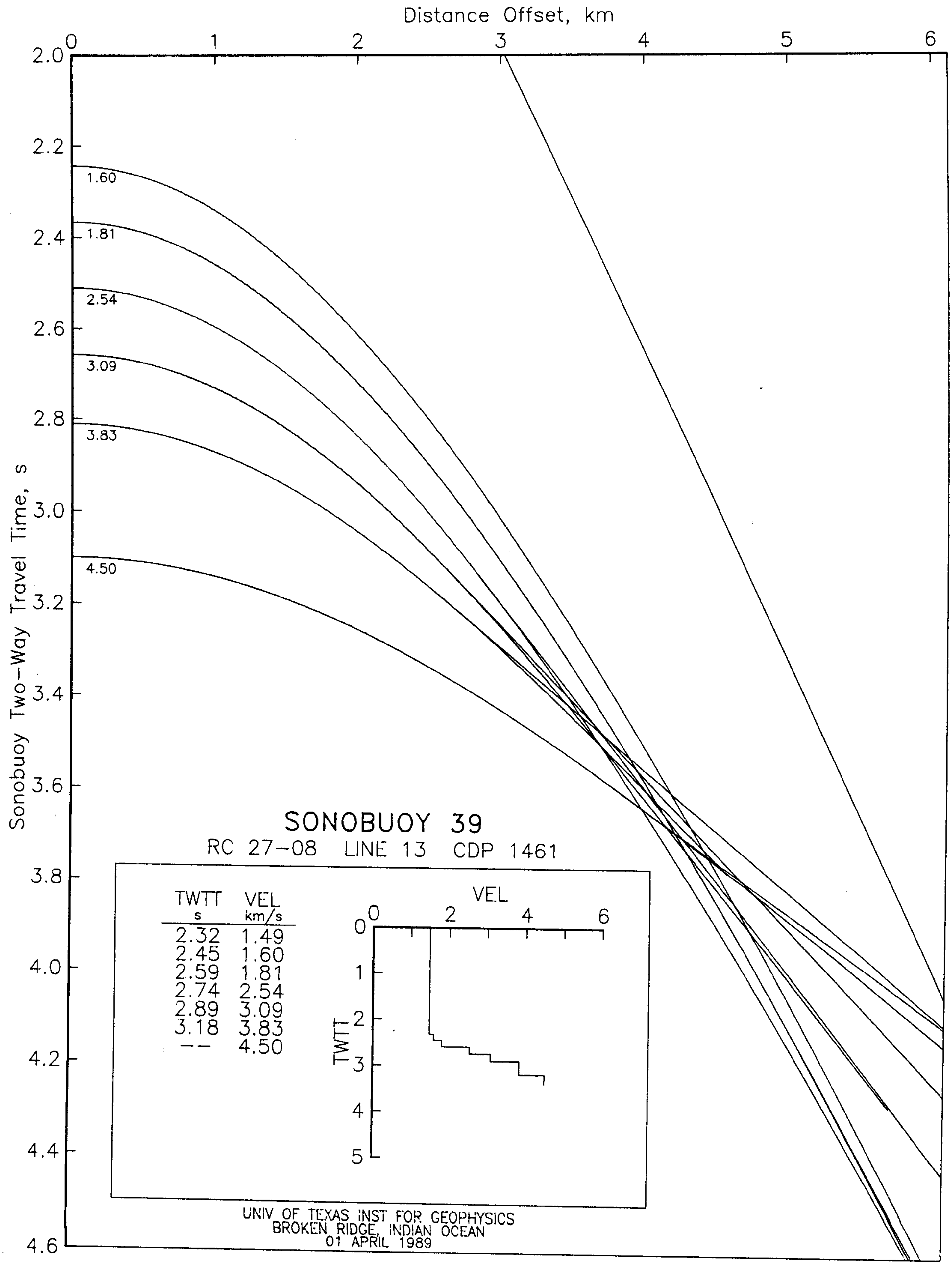

Distance Offset, km
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
Sonobuoy Two-Way Travel Time, s
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
4.8
5.0
5.2
1.60
2.21
3.55
4.30
SONOBUOY 34
RC 27-08 LINE 12 CDP 771
TWTT s
VEL km/s
2.88 1.49
3.13 1.60
3.31 2.21
3.47 3.55
-- 4.30
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

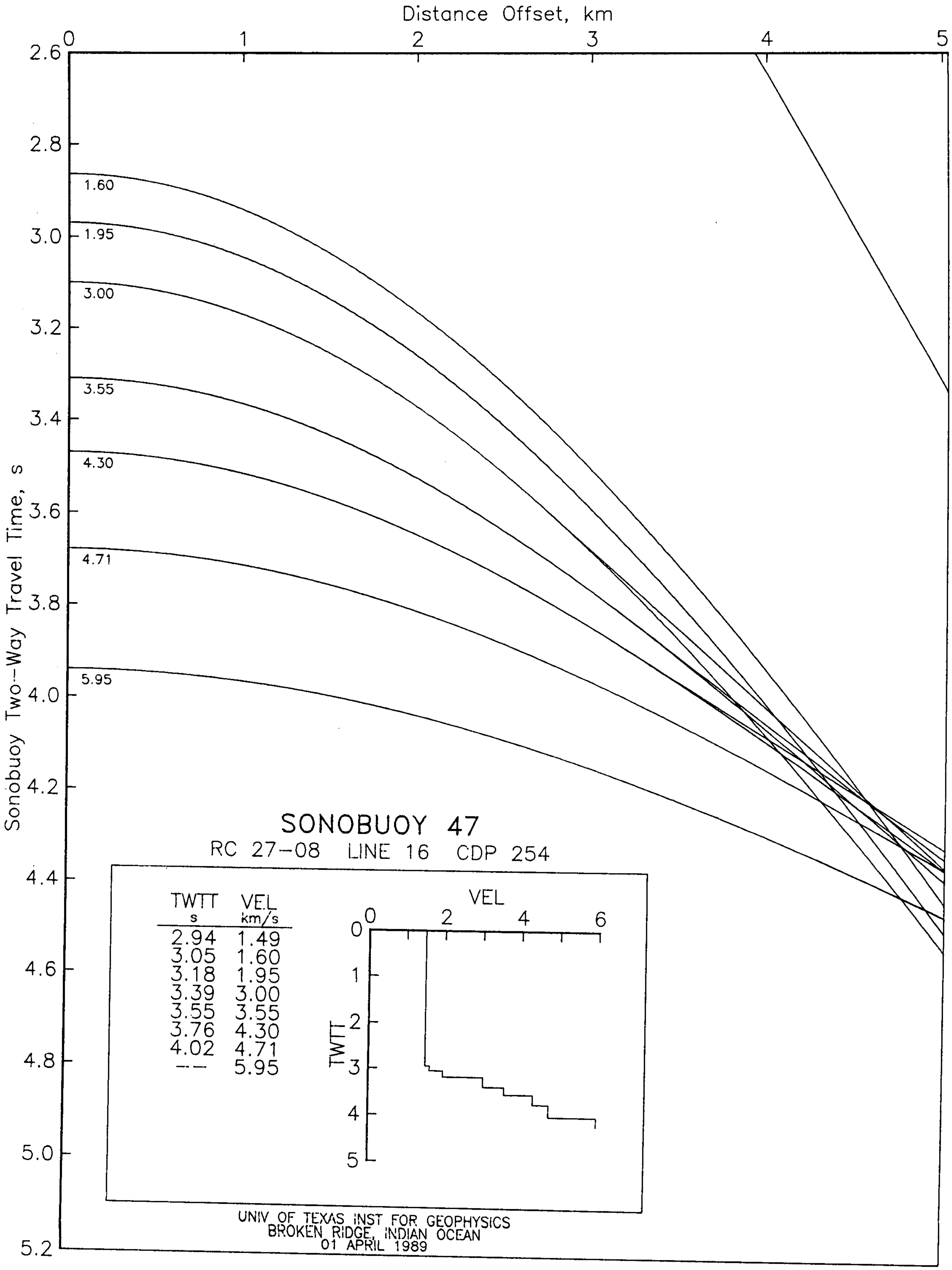


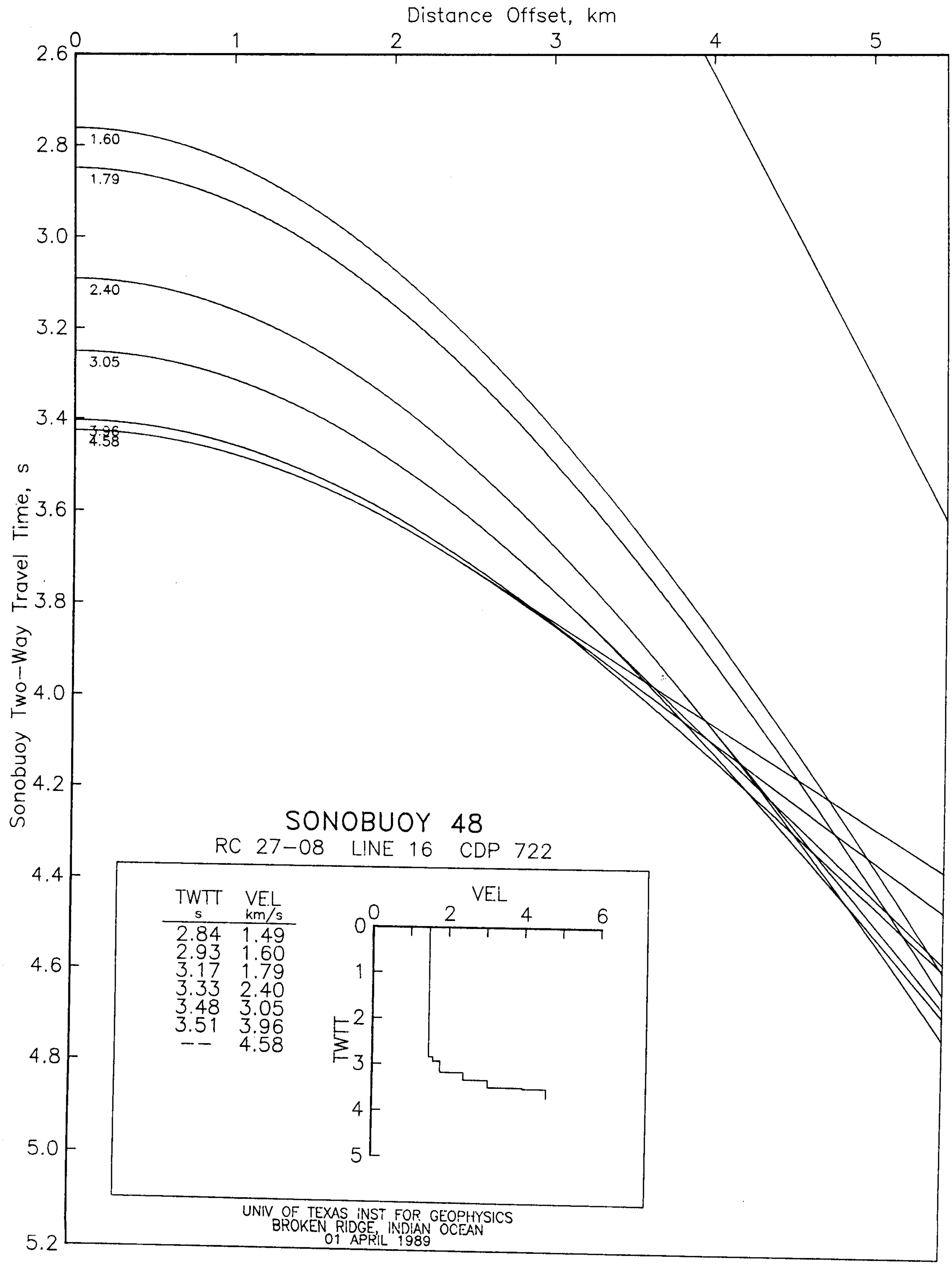
Distance Offset, km
0
1
2
3
4
5
6
Sonobuoy Two-Way Travel Time, s
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
4.8
5.0
5.2
1.60
1.70
2.43
3.09
3.90
SONOBUOY 35
RC 27-08 LINE 12 CDP 1375
TWTT s
VEL km/s
2.99 1.49
3.06 1.60
3.19 1.70
3.38 2.43
3.56 3.09
-·- 3.90
VEL
0
2
4
6
TWTT
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

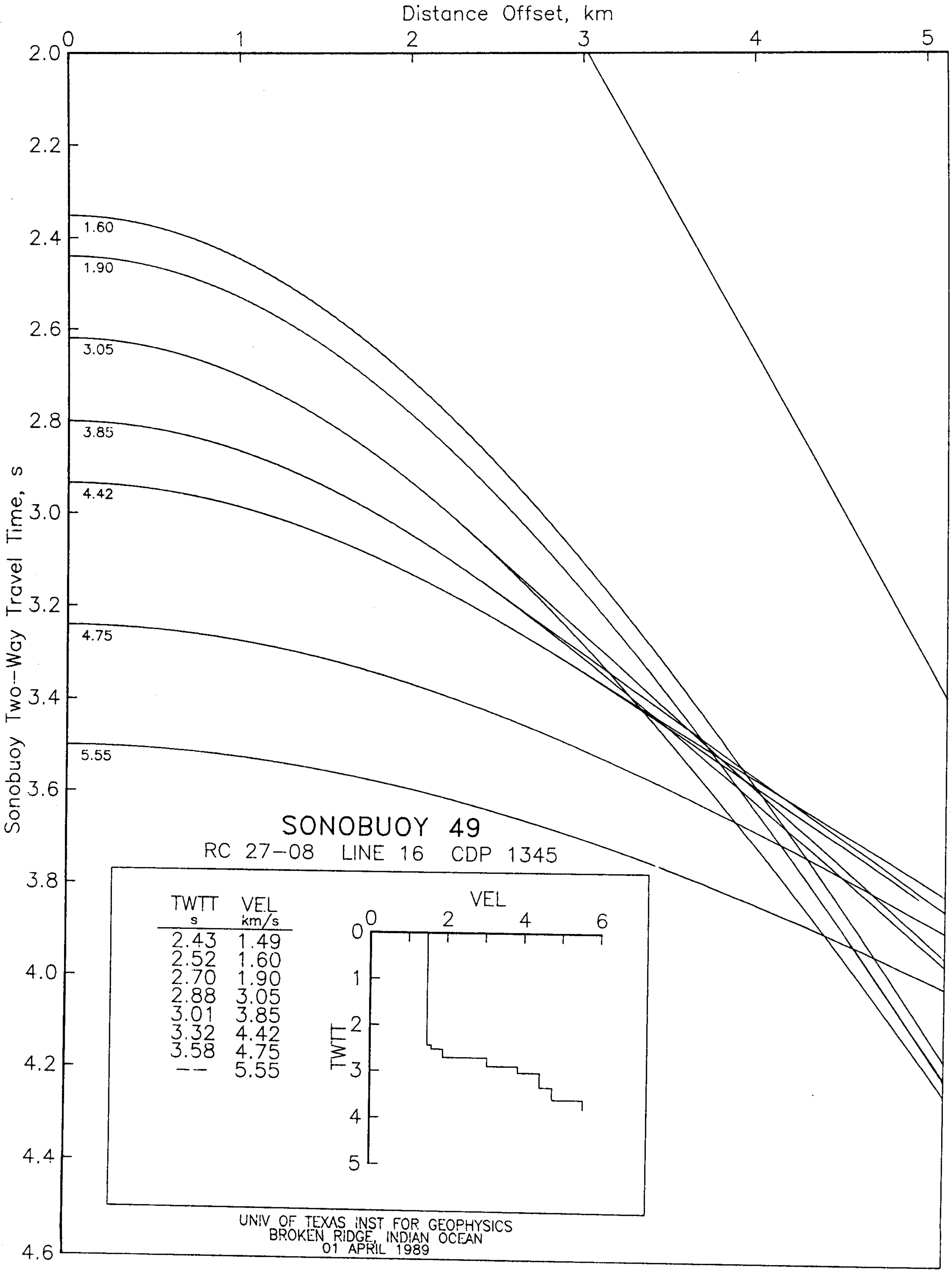

Distance Offset, km
0
1
2
3
4
5
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
Sonobuoy Two-Way Travel Time, s
1.60
1.70
2.25
2.95
3.83
SONOBUOY 36
RC 27-08 LINE 12 CDP 1642
TWTT s
VEL km/s
3.05 1.49
3.13 1.60
3.27 1.70
3.44 2.25
3.71 2.95
-.- 3.83
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

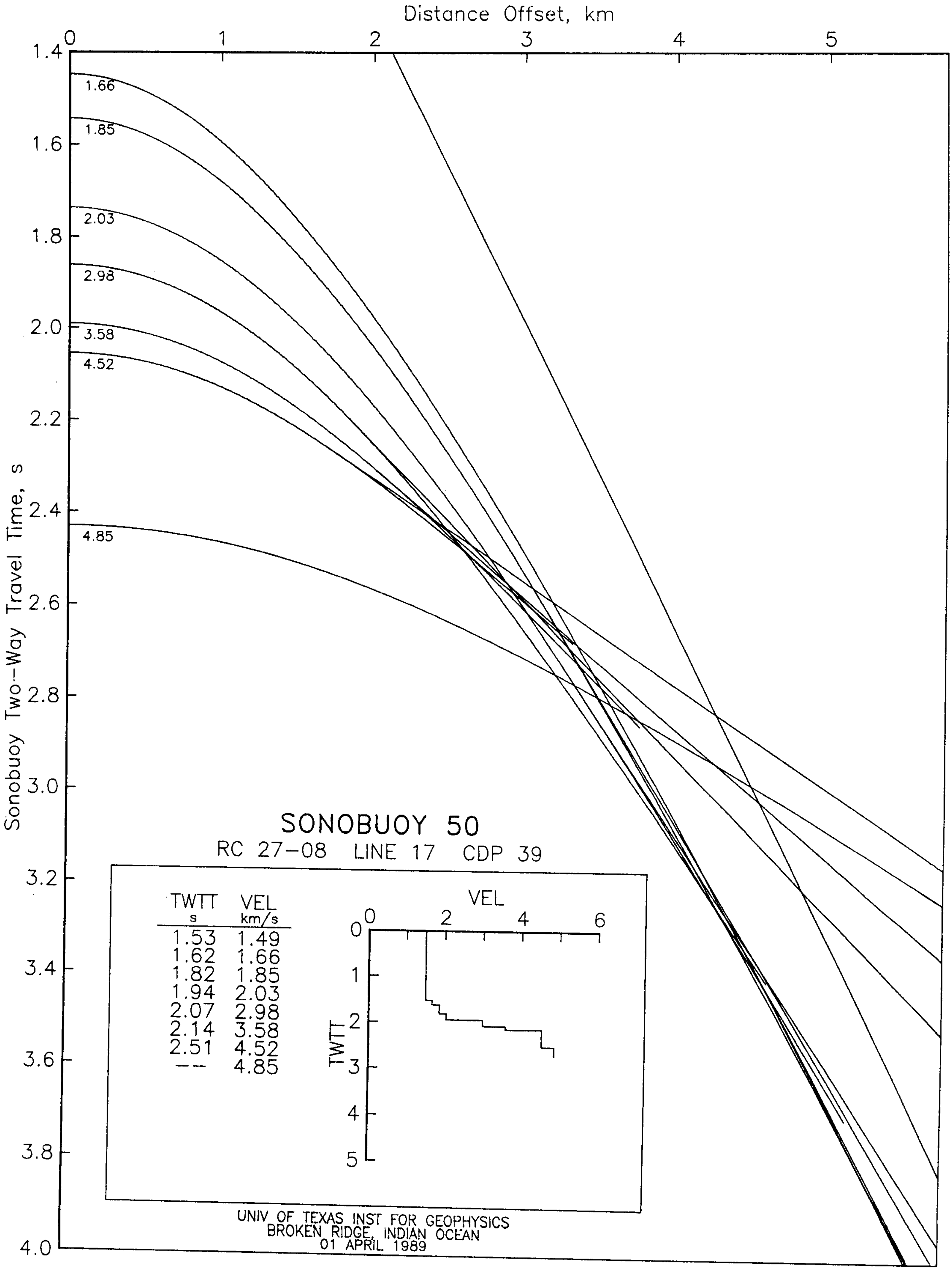

Distance Offset, km
0
1
2
3
4
5
6
Sonobuoy Two-Way Travel Time, s
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
4.8
5.0
5.2
1.60
1.85
2.21
3.09
3.91
4.45
4.84
SONOBUOY 38
RC 27-08 LINE 13 CDP 1160
TWTT s
VEL km/s
2.81 1.49
2.92 1.60
3.07 1.85
3.23 2.21
3.36 3.09
3.45 3.91
3.68 4.45
-- 4.84
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

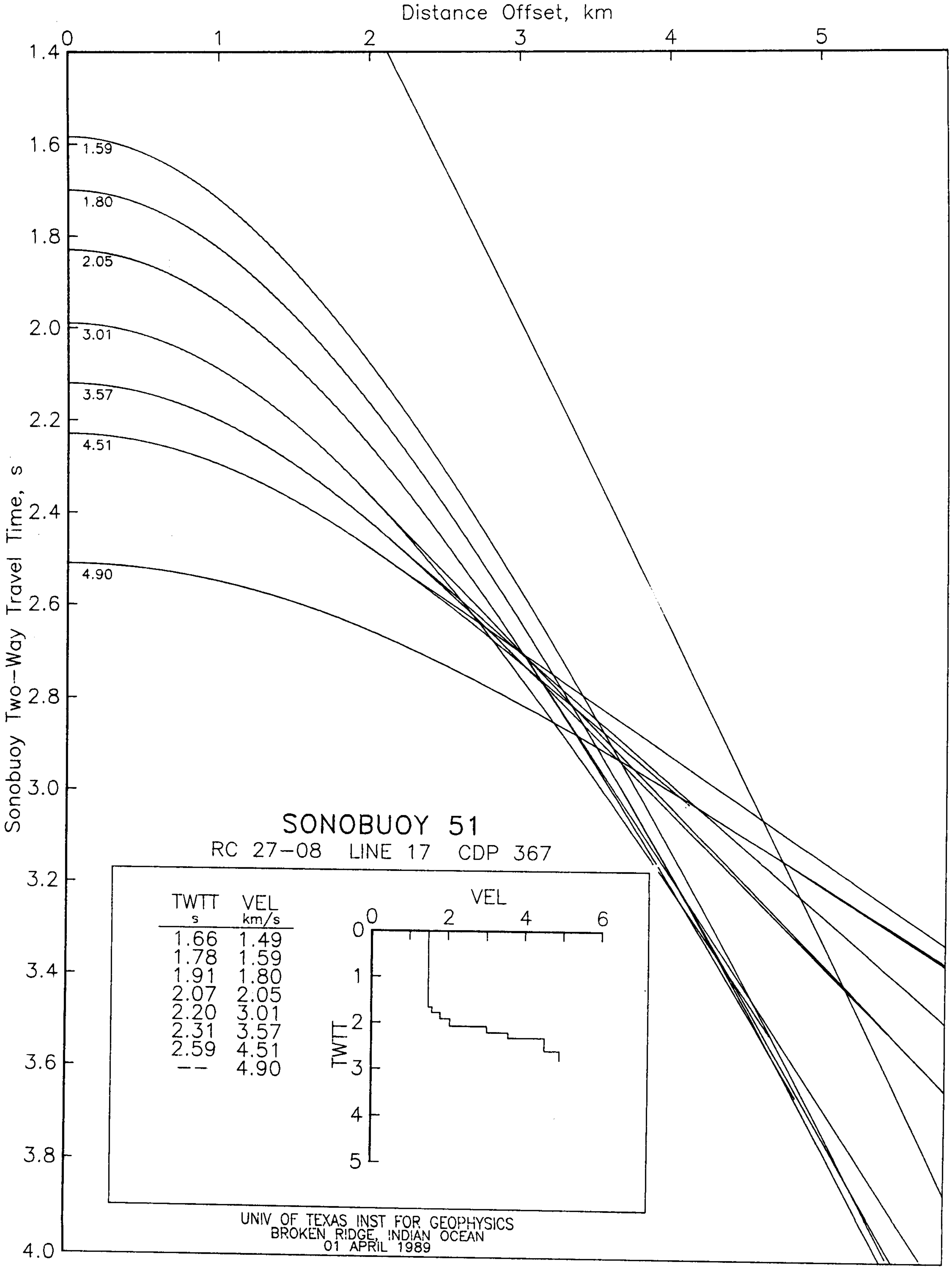

Distance Offset, km
0
1
2
3
4
5
6
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
Sonobuoy Two-Way Travel Time, s
1.60
1.81
2.54
3.09
3.83
4.50
SONOBUOY 39
RC 27-08 LINE 13 CDP 1461
TWTT s
VEL km/s
2.32 1.49
2.45 1.60
2.59 1.81
2.74 2.54
2.89 3.09
3.18 3.83
-- 4.50
VEL
TWTT
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

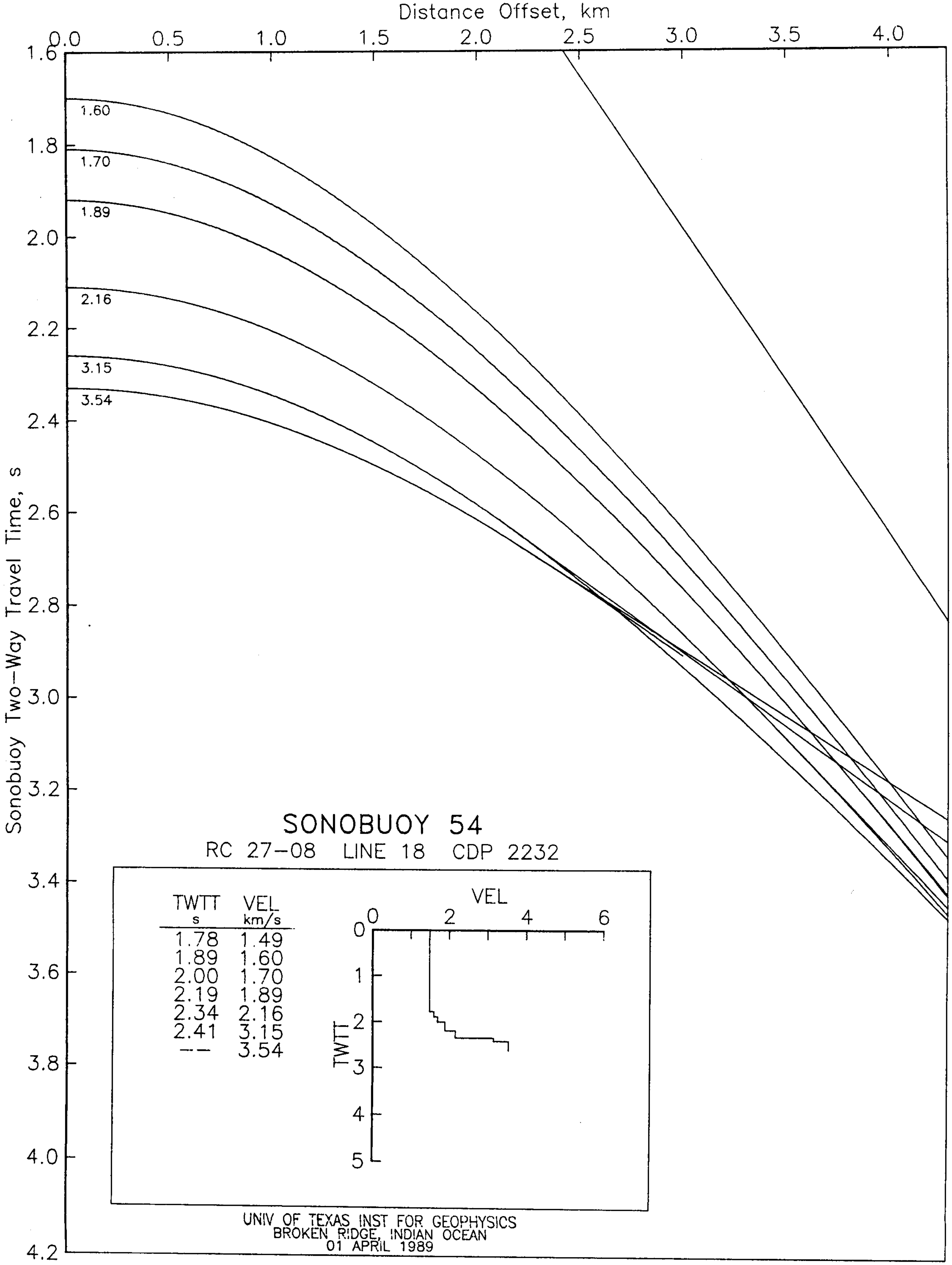

Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
4.8
5.0
5.2
1.60
1.95
3.00
3.55
4.30
4.71
5.95
SONOBUOY 47
RC 27-08 LINE 16 CDP 254
TWTT s
VEL km/s
2.94 1.49
3.05 1.60
3.18 1.95
3.39 3.00
3.55 3.55
3.76 4.30
4.02 4.71
-- 5.95
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

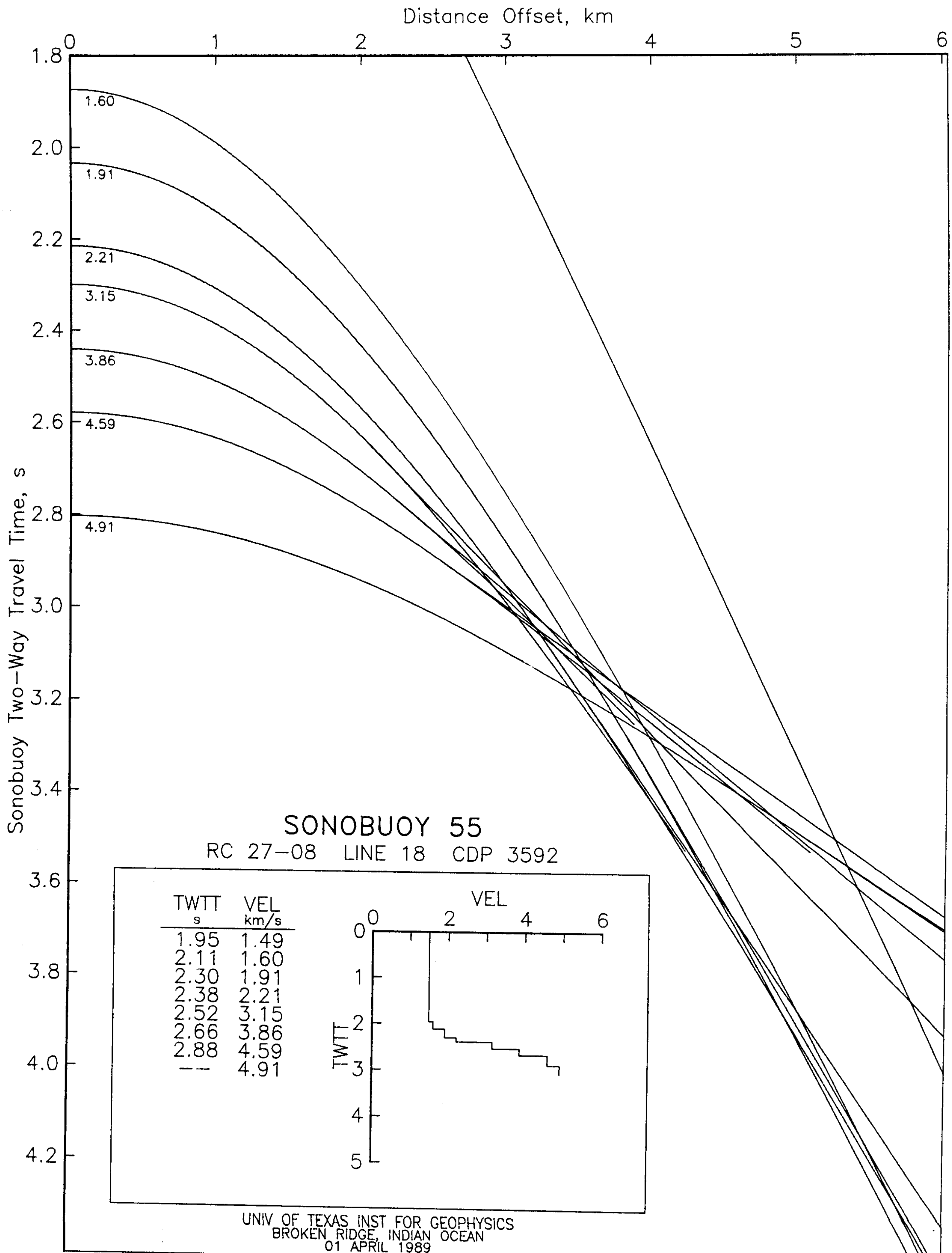

Distance Offset, km
Sonobuoy Two-Way Travel Time, s
1.60
1.79
2.40
3.05
3.96
4.58
SONOBUOY 48
RC 27-08 LINE 16 CDP 722
TWTT s
VEL km/s
2.84 1.49
2.93 1.60
3.17 1.79
3.33 2.40
3.48 3.05
3.51 3.96
-- 4.58
VEL
TWTT
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

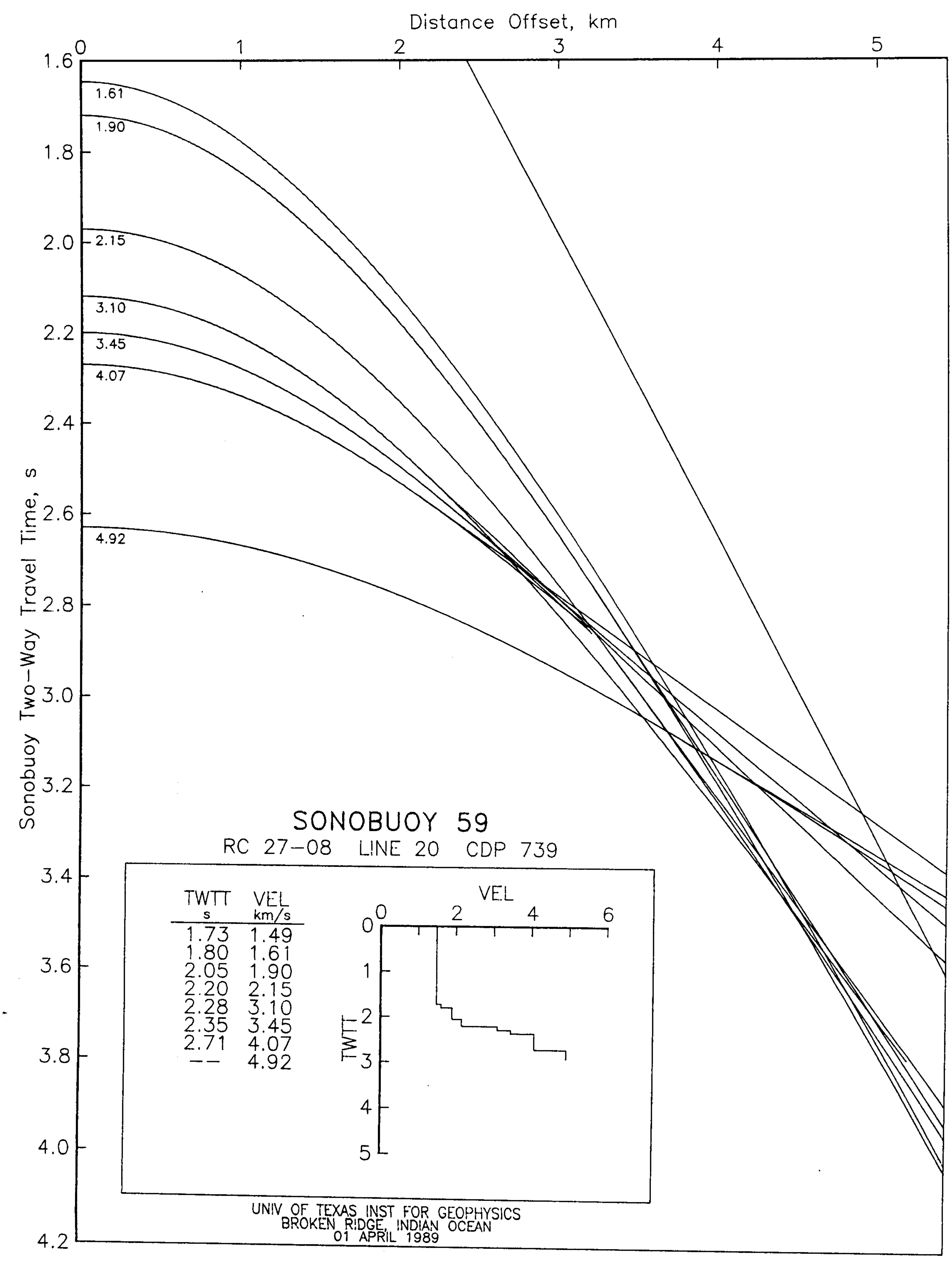

Distance Offset, km
0
1
2
3
4
5
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
4.4
4.6
Sonobuoy Two-Way Travel Time, s
1.60
1.90
3.05
3.85
4.42
4.75
5.55
SONOBUOY 49
RC 27-08 LINE 16 CDP 1345
TWTT s
VEL km/s
2.43 1.49
2.52 1.60
2.70 1.90
2.88 3.05
3.01 3.85
3.32 4.42
3.58 4.75
-- 5.55
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

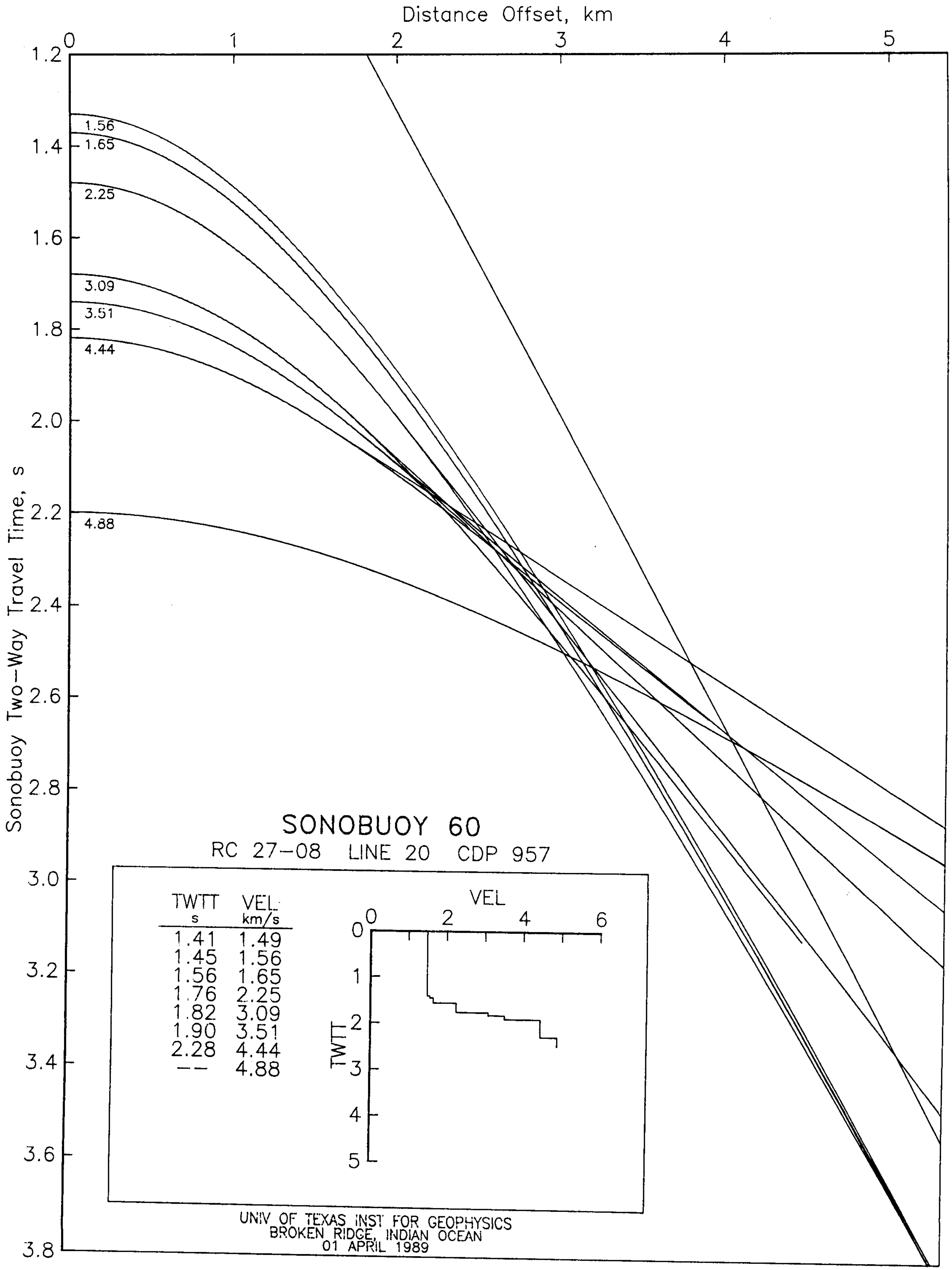

Distance Offset, km
Sonobuoy Two-Way Travel Time, s
1.66
1.85
2.03
2.98
3.58
4.52
4.85
SONOBUOY 50
RC 27-08 LINE 17 CDP 39
TWTT s
VEL km/s
1.53 1.49
1.62 1.66
1.82 1.85
1.94 2.03
2.07 2.98
2.14 3.58
2.51 4.52
--- 4.85
VEL
TWTT
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
1.59
1.80
2.05
3.01
3.57
4.51
4.90
SONOBUOY 51
RC 27-08 LINE 17 CDP 367
TWTT VEL
s km/s
1.66 1.49
1.78 1.59
1.91 1.80
2.07 2.05
2.20 3.01
2.31 3.57
2.59 4.51
-- 4.90
VEL
0 2 4 6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

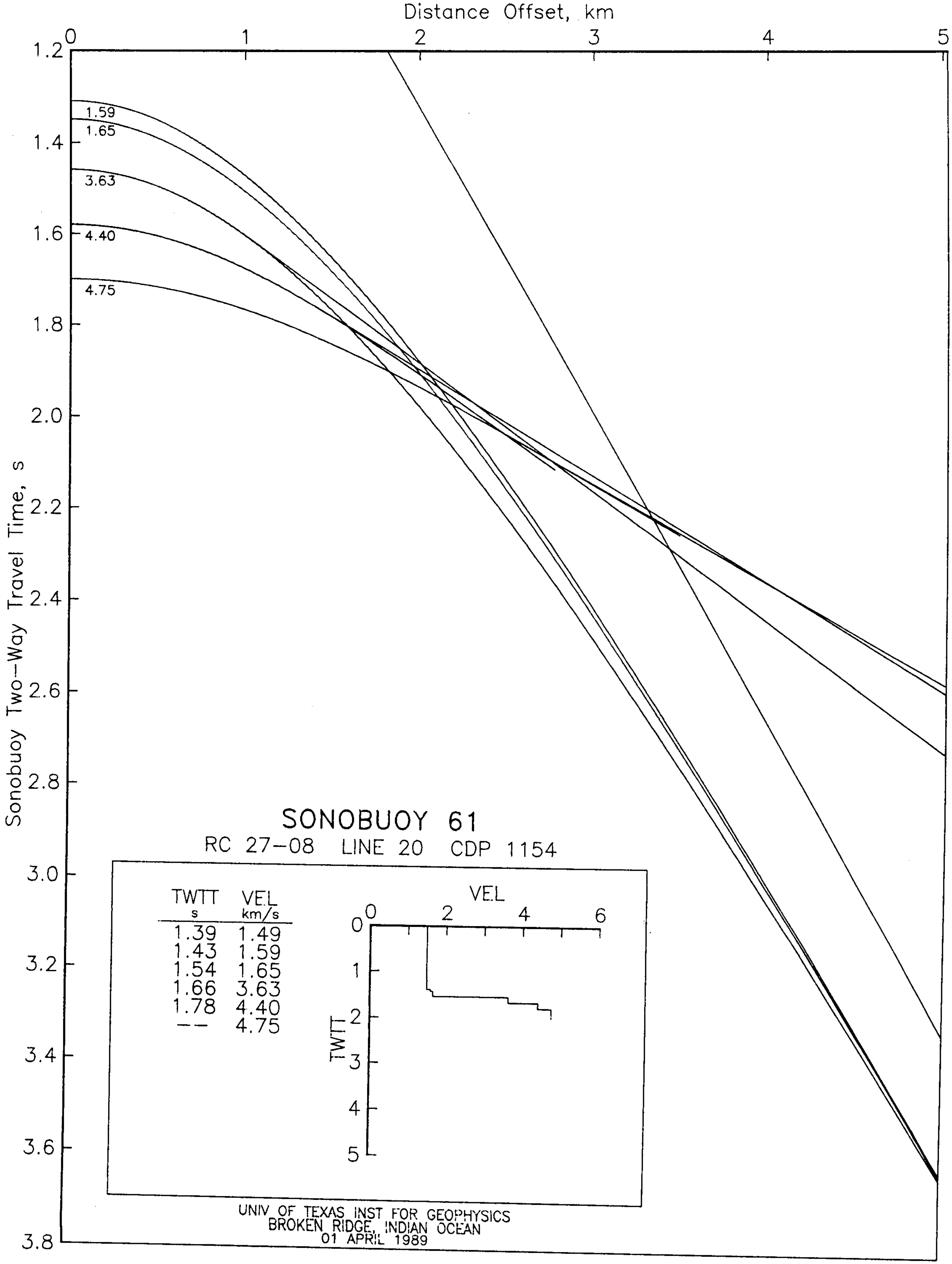

Distance Offset, km
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Sonobuoy Two-Way Travel Time, s
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
1.60
1.70
1.89
2.16
3.15
3.54
SONOBUOY 54
RC 27-08 LINE 18 CDP 2232
TWTT s
VEL km/s
1.78 1.49
1.89 1.60
2.00 1.70
2.19 1.89
2.34 2.16
2.41 3.15
-- 3.54
VEL
0 2 4 6
TWTT
0 1 2 3 4 5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
Sonobuoy Two-Way Travel Time, s
1.60
1.91
2.21
3.15
3.86
4.59
4.91
SONOBUOY 55
RC 27-08 LINE 18 CDP 3592
TWTT s | VEL km/s
1.95 1.49
2.11 1.60
2.30 1.91
2.38 2.21
2.52 3.15
2.66 3.86
2.88 4.59
-- 4.91
VEL
TWTT
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
4.2
1.61
1.90
2.15
3.10
3.45
4.07
4.92
SONOBUOY 59
RC 27-08 LINE 20 CDP 739
TWTT s
VEL km/s
1.73 1.49
1.80 1.61
2.05 1.90
2.20 2.15
2.28 3.10
2.35 3.45
2.71 4.07
-- 4.92
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989



Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
1.56
1.65
2.25
3.09
3.51
4.44
4.88
SONOBUOY 60
RC 27-08 LINE 20 CDP 957
TWTT s
VEL km/s
1.41 1.49
1.45 1.56
1.56 1.65
1.76 2.25
1.82 3.09
1.90 3.51
2.28 4.44
-- 4.88
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989


Distance Offset, km
0
1
2
3
4
5
Sonobuoy Two-Way Travel Time, s
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
1.59
1.65
3.63
4.40
4.75
SONOBUOY 61
RC 27-08 LINE 20 CDP 1154
TWTT s
VEL km/s
1.39 1.49
1.43 1.59
1.54 1.65
1.66 3.63
1.78 4.40
-- 4.75
VEL
0
2
4
6
TWTT
0
1
2
3
4
5
UNIV OF TEXAS INST FOR GEOPHYSICS
BROKEN RIDGE, INDIAN OCEAN
01 APRIL 1989

APPENDIX I -- Digitizing Software

```
      PROGRAM SONODIG
C
C...PROGRAM TO ESTIMATE ONE-DIMENSIONAL CONSTANT VELOCITY VS. DEPTH MODELS
C...FROM REFRACTED AND REFLECTED ARRIVALS OBSERVED IN T-X PLOTS OF SONOBUOY-
C...TYPE DATA.  USES DIGITIZED SLOPES AND TWO-WAY TRAVEL TIMES AS STARTING
C...ESTIMATES FOR AN ITERATIVE RAY TRACING SCHEME TO FIT PRECRITICAL AND
C...REFRACTED TRAVEL TIME BRANCHES THROUGH USER-SPECIFIED POINTS ON
C...OBSERVED ARRIVALS.  OUTPUT FROM THIS PROGRAM CAN BE USED WITH PROGRAM
C...PTXT TO GENERATE RAYTRACED TRAVEL TIME CURVES FOR GENERATION OF OVERLAY
C...PLOTS WITH ANY SCALING X-Y PLOT PROGRAM.
C...
C...THE T-X PLOTS USED FOR DIGITIZING MAY HAVE TIME AXES WHICH INCREASE IN
C...EITHER DIRECTION (+Y OR -Y) AND SOURCE-RECEIVER OFFSET AXES WHICH
C...INCREASE IN EITHER DIRECTION (+X OR -X).  IF THE SCALES OF THE TIME AND
C...DISTANCE AXES ARE KNOWN, THEY MAY BE INPUT AS CONSTANTS OR COMPUTED
C...FROM DIGITIZED AXIAL TIC MARKS.  IF THE DISTANCE AXIS SCALE IS UNKNOWN,
C...IT MAY BE COMPUTED AUTOMATICALLY FROM DIGITIZED POINTS ON THE DIRECT
C...WATER WAVE ARRIVAL.  THIS PROGRAM WILL ALSO ACCOMODATE VELOCITY-REDUCED
C...PLOTS.
C...
C...THE MODEL USED WILL ALLOW FOR SOURCE AND/OR RECEIVER TO BE BURIED
C...IN THE FIRST LAYER AND PROVIDES FOR SEPARATE LAYER 1, NEAR-SOURCE
C...AND NEAR-RECEIVER VELOCITIES TO BE DECLARED FOR INCREASED ACCURACY.
C...OUTPUTS LAYER VELOCITIES, VERTICAL TWO-WAY TRAVEL TIMES FOR BOTH
C...SEA-SURFACE AND BURIED RECEIVERS, LAYER DEPTHS AND LAYER THICKNESSES.
C...
C...AFTER DETERMINING THE PLOT SCALES, THE BASIC PROCEDURE IS TO:
C...    1) INPUT THE VELOCITY OF THE NEXT LAYER BY DIGITIZING A SLOPE IN X-T
C...       OR BY PROVIDING AN EXTERNALLY DERIVED ESTIMATE.
C...    2) INPUT AN ESTIMATE OF THE TWO-WAY TRAVEL TIME (AS DETERMINED BY THE
C...       BURIED RECEIVER, IE., THE SONOBUOY) TO THE BOTTOM OF THE CURRENT
C...       LAYER (THE TOP OF THE NEXT LAYER).
C...    3) INPUT A REFERENCE POINT IN X-T WHICH LIES ON THE TRAVEL TIME CURVE
C...       DESCRIBING THE PRE-CRITICAL REFLECTION OFF THE BOTTOM OF THE
C...       CURRENT LAYER (THE TOP OF NEXT LAYER) OR THE REFRACTION FROM JUST
C...       BELOW THIS INTERFACE (IE. IN THE NEXT LAYER)..
C...    4) USE A 1-D RAYTRACING ALGORITHM TO COMPUTE THE TRAVEL TIME CURVES
C...       FOR THE PRECRITICAL REFLECTION OFF THE BOTTOM OF THE CURRENT
C...       LAYER (THE TOP OF THE NEXT LAYER) AND FOR THE REFRACTION (HEAD
C...       WAVE) FROM THE LAYER BELOW.
C...    5) DETERMINE WHETHER THE TEST POINT LIES ON THE COMPUTED TRAVEL TIME
C...       CURVES TO WITHIN THE ALLOWED TRAVEL TIME MISFIT ERROR.  IF NOT,
C...       PERTURB THE VERTICAL TWO-WAY TRAVEL TIME ESTIMATE (IE., THE
C...       THICKNESS OF THE CURRENT LAYER) AND REPEAT STEPS 4 AND 5.  ONCE A
C...       FIT SATISFACTORY TO THE USER IS ACHIEVED, PROCEED TO STEP 1 AND
C...       REPEAT THE SEQUENCE FOR THE NEXT LAYER DOWN.
C...THE USER MAY PROVIDE KEYBOARD ENTRIES IN LIEU OF DIGITIZING POINTS FOR
C...STEPS 1-3.  NOTE THAT THE SLOPE IN STEP 1 AND THE TWTT IN STEP 2 MAY BE
C...DIGITIZED FROM ANY CONVENIENT LOCATION ON THE PLOT AND NEED NOT LIE ON
C...A SPECIFIC TRAVEL TIME CURVE.  MODEL LAYERS MAY ALSO BE INPUT FROM A
C...FILE ALLOWING THE USER TO SUBDIVIDE AN X-T PLOT COMPLICATED WITH
C...DISCONTINUITES AND SCALE CHANGES IN THE LINEAR X-AXIS (E.G., DUE TO
C...SHOOTING SHIP SPEED CHANGES) AND SUCCESSFULLY BUILD A MODEL BY
C...MULTIPLE PASSES THROUGH THE PROGRAM.
C...
C...THIS PROGRAM WILL WORK ON A VARIETY OF 'SMART' AND 'DUMB' DIGITIZERS.
```

```
C...AND CAN PERFORM PLOT-LEVELING THROUGH SOFTWARE ROTATIONS WHEN FIRMWARE
C...OR HARDWARE ROTATION FUNCTIONS ARE EITHER UNAVAILABLE OR UNDESIREABLE.
C...
C...COPYRIGHT 1988 BY KEVIN MACKENZIE, UNIV OF TEXAS INSTITUTE FOR GEOPHYSICS
C...
C...THIS PROGRAM AND ITS SUBROUTINES ARE FREEWARE AND MAY BE COPIED,
C...DISTRIBUTED AND USED FREELY, BUT CANNOT BE SOLD OR MODIFIED WITHOUT
C...EXPLICIT WRITTEN CONSENT FROM THE AUTHOR.
C...THE AUTHOR MAKES NO WARRANTIES OR GUARANTEES EXPRESS OR IMPLIED
C...AS TO THE ACCURACY OF THIS CODE OR AS TO THE CONSEQUENCES OF ITS USE.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS ROUTINES: DELTAU, DIGINI, DIGIT, DIGROT, DIGEND
C...
      CHARACTER NAME1*80,NAME2*80,ANS*1,BUTTON*1
      REAL*4 V(0:20),TWT(0:20),H(20),Z(0:20)
      REAL*4 DTAU(200),TAU(201),P(201),T(200),X(200),TAUSUM(201)
      Z(0)=0.
      NDEV=1       !DIGITIZING DEVICE: 1=CALCOMP9600,2=SUMMA36x48,3=SUMMA12x18
      VRED=6.      !REDUCING VELOCITY OF SEISMIC PLOT (VRED=0.0=INFINITY)
      ZRCV=.018    !RECEIVER DEPTH  .122KM=400FT, .018KM=60FT
      ZSRC=0.      !SOURCE DEPTH
      VRCV=1.495    !VELOCITY NEAR RECEIVER
      VSRC=1.495    !VELOCITY NEAR SOURCE
      VW=1.495      !VELOCITY WATER COLUMN
      NRAY=100     !MAXIMUM NUMBER OF RAYS TRACED
      XMAX=50.     !MAXIMUM DISTANCE OUT TO WHICH RAYS ARE TRACED
      DELT=.001    !TOLERATED TRAVEL TIME MISMATCH
      NSTEP=10     !MAXIMUM NUMBER OF ITERATIONS PER LAYER
      WRITE (*,'(40X,A)')     'SONODIG 3.0'
      WRITE (*,'(40X,A///)') 'Copyright 1988 by K. MacKenzie'
C
C GET DIGITIZING DEVICE PARAMETERS AND INITIALIZE DIGITIZER
      WRITE (*,'(A,A/A)')' AVAILABLE DIGITIZING DEVICES:',
     .' 1=CALCOMP9600, 2=SUMMA36x48, 3=SUMMA12x18.',
     .'                                ^DEFAULT^'
      WRITE (*,'(A)')' NDEV='
      IDEV=0
      READ (*,'(I1)') IDEV
      IF (IDEV.LT.1 .OR. IDEV.GT.3) IDEV=NDEV
      NDEV=IDEV
      CALL DIGINI (NDEV)
C
C GET PROGRAM PARAMETERS
    5 WRITE (*,'(/A,F6.3,A)')' ZSRC= ',ZSRC,
     .' KM          (SOURCE DEPTH)'
      WRITE (*,'(A,F6.3,A)')' ZRCV= ',ZRCV,
     .' KM          (RECEIVER DEPTH .122=400 ft, .018=60 ft)'
      WRITE (*,'(A,F6.3,A)')' VSRC= ',VSRC,
     .' KM/S        (VELOCITY NEAR SOURCE)'
      WRITE (*,'(A,F6.3,A)')' VRCV= ',VRCV,
     .' KM/S        (VELOCITY NEAR RECEIVER)'
      WRITE (*,'(A,F6.3,A)')' VW=   ',VW,
     .' KM/S        (VELOCITY OF WATER COLUMN)'
      WRITE (*,'(A,F6.3,A)')' VRED= ',VRED,
     .' KM/S        (REDUCING VELOCITY OF PLOT,VRED=0.0=INFINITY)'
      WRITE (*,'(/A)')' CHANGE ZSRC, ZRCV, VSRC, VRCV, VW OR VRED? [N] '
```

```
      READ (*,'(A1)') ANS
      IF (ANS.EQ.'Y' .OR. ANS.EQ.'y') THEN
         WRITE (*,'(/A)')' ZSRC='
         READ (*,*) ZSRC
         WRITE (*,'(A)')' ZRCV='
         READ (*,*) ZRCV
         WRITE (*,'(A)')' VSRC='
         READ (*,*) VSRC
         WRITE (*,'(A)')' VRCV='
         READ (*,*) VRCV
         WRITE (*,'(A)')' VW=  '
         READ (*,*) VW
         WRITE (*,'(A)')' VRED='
         READ (*,*) VRED
         IF (VRED.LT.0.)  VRED=0.
         IF (VRED.GT.50.) VRED=0.
         GO TO 5
      END IF
      IF (VSRC.LE.0. .OR. VRCV.LE.0. .OR. VW.LE.0.) THEN
         WRITE (*,'(/A)')' *** VELOCITIES MUST BE .GT. ZERO.'
         GO TO 5
      END IF
      IF (ZSRC.LT.0.) ZSRC=0.
      IF (ZRCV.LT.0.) ZRCV=0.
      IF (VRED.NE.0.) THEN
         PRED=1./VRED
      ELSE
         PRED=0.
      END IF
   10 WRITE (*,'(/A,F5.1,A)')' XMAX= ',XMAX,
     .' KM          (MAX X OF RAYTRACE)'
      WRITE (*,'(A,I4,A)')  ' NRAY= ',NRAY,
     .'             (NUMBER OF RAYS TRACED)'
      WRITE (*,'(A,F6.4,A)')' DELT= ',DELT,
     .' S          (MAX TIME MISFIT ERROR)'
      WRITE (*,'(A,I3,A)')  ' NSTEP=',NSTEP,
     .'              (MAX NUMBER OF ITERATIONS)'
      WRITE (*,'(/A)') ' CHANGE XMAX, NRAY, DELT OR NSTEP? [N] '
      READ (*,'(A1)') ANS
      IF (ANS.EQ.'Y' .OR. ANS.EQ.'y') THEN
         WRITE (*,'(A)')' XMAX='
         READ (*,*) XMAX
         WRITE (*,'(A)')' NRAY='
         READ (*,*)    NRAY
         WRITE (*,'(A)')' DELT='
         READ (*,*) DELT
         WRITE (*,'(A)')' NSTEP='
         READ (*,*)    NSTEP
         GO TO 10
      END IF
      IF (XMAX.LE.0.0 .OR. DELT.LE.0.0 .OR. NSTEP.LE.0) THEN
         WRITE (*,'(/A)')' *** XMAX, DELT AND NSTEP MUST BE .GT. ZERO.'
         GO TO 10
      ELSE IF (NRAY.LT.2 .OR. NRAY.GT.200) THEN
         WRITE (*,'(/A)')' *** NRAY MUST BE .GE. 2 AND .LE. 200'
         GO TO 10
      END IF
      WRITE (*,'(/A)') ' NAME OF DIGITIZED OUTPUT FILE? '
```

```
      READ (*,'(A80)') NAME1
C
C CHECK FOR PLOT ALIGNMENT
   20 PRINT *,' '
      PRINT *,' VERIFY PLOT ALIGNMENT--DIGITIZE TWO POINTS ALONG ANY',
     .' HORIZONTAL LINE:'
      CALL DIGIT (BUTTON,X1,Y1)
      WRITE (*,'(20X,A,F7.3,A,F7.3,A)')
     .' X1=',X1,'  Y1=',Y1,'  BUTTON='//BUTTON
      CALL DIGIT (BUTTON,X2,Y2)
      WRITE (*,'(20X,A,F7.3,A,F7.3,A)')
     .' X2=',X2,'  Y2=',Y2,'  BUTTON='//BUTTON
      DY=ABS(Y2-Y1)
      IF (DY.GT.0.05) THEN
         PRINT *,' '
         WRITE (*,'(/A,A/A/A,A/A)')
     .   ' *** X-AXIS OF PLOT TILTED MORE THAT 0.05 INCHES RELATIVE TO',
     .     ' DIGITIZING AXIS.',
     .   '       EITHER MANUALLY REALIGN PAPER ON DIGITIZER AND RETEST,',
     .   '       OR USE EXTERNAL HARDWARE ROTATION FUNCTION ON THE',
     .     ' DIGITIZER AND RETEST,',
     .   '       OR SELECT SOFTWARE PLOT ROTATION.'
   30    PRINT *,' '
         PRINT *,' SELECT SOFTWARE PLOT ROTATION WITH BUTTON "1"'
         PRINT *,' OR SELECT RETEST ALIGNMENT WITH BUTTON "2"'
         PRINT *,' OR DIGITIZE AND PRINT A TEST POINT WITH BUTTON "3":'
         CALL DIGIT (BUTTON,XDUM,YDUM)
         IF (BUTTON.EQ.'1') THEN
            WRITE (*,'(/A)') ' PERFORMING SOFTWARE ROTATION OF PLOT.'
            CALL DIGROT (X1,Y1,X2,Y2)
            GO TO 40
         ELSE IF (BUTTON.EQ.'2') THEN
            GO TO 20
         END IF
         WRITE (*,'(/20X,A,F7.3,A,F7.3,A)')
     .   ' X=',XDUM,'  Y=',YDUM,'  BUTTON='//BUTTON
         GO TO 30
      END IF
C
C OPEN OUTPUT FILE
      OPEN (UNIT=10,FILE=NAME1,STATUS='NEW')
C
C SET UP LIST OF RAYS (P=0.0 TO P=UWR) AND ZERO OUT TAUSUM
   40 USRC=1./VSRC
      URCV=1./VRCV
      UW=1./VW
      UWR=AMAX1(UW,USRC,URCV)
      DO 45 I=1,NRAY+1
         TAUSUM(I)=0.0
         P(I)=UWR*REAL(I-1)/REAL(NRAY-1)
   45 CONTINUE
C
C PRECORRECT TAUSUM FOR SOURCE BURIED IN FIRST LAYER
      IF (ZSRC.GT.0.) THEN
         HSRC=ZSRC*0.5
         CALL DELTAU (VSRC,HSRC,NRAY,P,TAU,NDT)
         DO 50 I=1,NDT
            TAUSUM(I)=-TAU(I)
```

```
   50    CONTINUE
      END IF
      TSRC=ZSRC*UW
C
C PRECORRECT TAUSUM FOR RECEIVER BURIED IN FIRST LAYER
      IF (ZRCV.GT.0.) THEN
         HRCV=ZRCV*0.5
         CALL DELTAU (VRCV,HRCV,NRAY,P,TAU,NDT)
         DO 60 I=1,NDT
            TAUSUM(I)=TAUSUM(I)-TAU(I)
   60    CONTINUE
      END IF
      TRCV=ZRCV*UW
C
C INITIALIZE LAYER INDICES
      K=0
      K1=K+1
C WANT TWT(1)-TWT(0) TO EQUAL TOTAL LAYER TT
      TWT(K)=-(TRCV+TSRC)
      V(K)=0.
      V(K1)=VW
      UK1=UW
C
C COMPUTE DIRECT WATER WAVE TRAVEL TIME TABLE ACCOUNTING FOR ANY DIFFERENCE
C IN SOURCE AND RECEIVER DEPTHS.  NOTE THAT TRAVEL TIMES ARE STORED AS
C REDUCED TRAVEL TIMES.
   70 ZDIF=ABS(ZRCV-ZSRC)
      VMEAN=(VRCV+VSRC)*0.5
      UMEAN=1./VMEAN
      IF (ZDIF.GT.0.) THEN
         HDIF=ZDIF*0.5
         CALL DELTAU (VMEAN,HDIF,NRAY,P,TAU,NDT)
         X(1)=0.0
         T(1)=TAU(1)
         TAU(NDT+1)=TAU(NDT) + (TAU(NDT)-TAU(NDT-1))
         DO 80 I=2,NDT
            X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
            T(I)=(P(I)-PRED)*X(I) + TAU(I)
   80    CONTINUE
      ELSE
         X(1)=0.0
         X(2)=XMAX
         T(1)=0.0
         T(2)=X(2)*(UMEAN-PRED)
         NDT=2
      END IF
C
C GET KNOWN SCALE INFO FROM USER
      XSCL=0.0
      WRITE (*,'(/A)')
     .' TYPE IN X-SCALE IN  IN/KM (0.0, IF UNKNOWN):  [0.0] '
      READ (*,'(F10.0)') XSCL
      IF (XSCL.NE.0.) THEN
         XSCL=1./XSCL
      END IF
      TSCL=0.0
      WRITE (*,'(/A)')
     .' TYPE IN T-SCALE IN  IN/SEC (0.0, IF UNKNOWN):  [0.0] '
```

```
      READ (*,'(F10.0)') TSCL
      IF (TSCL.NE.0.) THEN
         TSCL=1./TSCL
      END IF
C
C DIGITIZE UNKNOWN SCALE INFO
      IF (XSCL.EQ.0. .AND. TSCL.EQ.0.) THEN
         WRITE (*,'(/A,A)')
     .   ' DIGITIZE TWO PTS, T1 AND T2, ON THE DIRECT WATER WAVE',
     .   ' ARRIVAL: (T2>T1)'
         CALL DIGIT (BUTTON,X1,Y1)
         CALL DIGIT (BUTTON,X2,Y2)
         WRITE (*,'(/A)')
     .   ' TYPE IN THE TRAVEL TIMES (SEC) T1,T2 FOR BOTH PTS: '
         READ (*,*) T1,T2
      ELSE IF (TSCL.EQ.0.) THEN
         WRITE (*,'(/A,A)')
     .   ' DIGITIZE TWO PTS, (X1,T1) AND (X1,T2), ON A LINE PARALLEL',
     .   ' TO THE TIME AXIS (T2>T1):'
         CALL DIGIT (BUTTON,X1,Y1)
         CALL DIGIT (BUTTON,X2,Y2)
         WRITE (*,'(/A)')
     .   ' TYPE IN X (KM) AND TIME (SEC) VALUES FOR X1, T1 AND T2: '
         READ (*,*) XP1,T1,T2
      ELSE IF (XSCL.EQ.0.) THEN
         WRITE (*,'(/A,A)')
     .   ' DIGITIZE TWO PTS, (X1,T1) AND (X2,T1), ON A LINE PARALLEL',
     .   ' TO THE X AXIS (X2>X1):'
         CALL DIGIT (BUTTON,X1,Y1)
         CALL DIGIT (BUTTON,X2,Y2)
         WRITE (*,'(/A)')
     .   ' TYPE IN X (KM) AND TIME (SEC) VALUES FOR X1, T1 AND X2: '
         READ (*,*) XP1,T1,XP2
      ELSE
         WRITE (*,'(/A)')
     .   ' DIGITIZE ONE PT, (X1,T1), AS AN ORIGIN REFERENCE:'
         CALL DIGIT (BUTTON,X1,Y1)
         WRITE (*,'(/A)')
     .   ' TYPE IN X (KM) AND TIME (SEC) VALUES FOR X1 AND T1: '
         READ (*,*) XP1,T1
      END IF
      PRINT *,' '
C
C INTERPOLATE OFFSETS OF DIGITIZED DIRECT WATER WAVE TRAVEL TIMES
      IF (XSCL.EQ.0. .AND. TSCL.EQ.0.) THEN
         DO 200 I=1,NDT-1
             IF (T1.GE.T(I).AND.T1.LE.T(I+1)) THEN
                XW1=(X(I+1)-X(I))*(T1-T(I))/(T(I+1)-T(I)) + X(I)
                GO TO 210
             END IF
  200    CONTINUE
         XW1=(T1-T(NDT))/(UMEAN-PRED) + X(NDT)
  210    DO 220 I=1,NDT-1
             IF (T2.GE.T(I).AND.T2.LE.T(I+1)) THEN
                XW2=(X(I+1)-X(I))*(T2-T(I))/(T(I+1)-T(I)) + X(I)
                GO TO 250
             END IF
  220    CONTINUE
```

```
         XW2=(T2-T(NDT))/(UMEAN-PRED) + X(NDT)
      ELSE IF (XSCL.NE.0. .AND. TSCL.NE.0.) THEN
         DO 230 I=1,NDT-1
             IF (X1.GE.X(I).AND.X1.LE.X(I+1)) THEN
                TW1=(T(I+1)-T(I))*(X1-X(I))/(X(I+1)-X(I)) + T(I)
                GO TO 240
             END IF
  230    CONTINUE
         TW1=(X1-X(NDT))*(UMEAN-PRED) + T(NDT)
  240    WRITE (*,'(20X,A,F8.4,A,F7.4,A,F7.4,A)')
     .   ' T1(X1)-TW(X1)=',T1-TW1,' S   VSRC=',VSRC,
     .   ' KM/S  VRCV=',VRCV,' KM/S'
      END IF
C
C DETERMINE SCALES
  250 IF (XSCL.EQ.0. .AND. TSCL.EQ.0.) THEN
C ...NO SCALE GIVEN SO DETERMINE BOTH X- AND T-SCALES FROM DIRECT WATER WAVE
         X0=X1
         Y0=Y1
         DT=T2-T1
         DXW=XW2-XW1
         DX=X2-X1
         DY=Y1-Y2
         TSCL=DT/DY
         XSCL=DXW/DX
         TREF=T1
         XREF=XW1
      ELSE IF (TSCL.EQ.0.) THEN
C ...X-SCALE GIVEN SO DETERMINE T-SCALE
         X0=X1
         Y0=Y1
         DT=T2-T1
         DY=Y1-Y2
         TSCL=DT/DY
         TREF=T1
         XREF=XP1
      ELSE IF (XSCL.EQ.0.) THEN
C ...T-SCALE GIVEN SO DETERMINE X-SCALE
         X0=X1
         Y0=Y1
         DXP=XP2-XP1
         DX=X2-X1
         XSCL=DXP/DX
         TREF=T1
         XREF=XP1
      ELSE
C ...BOTH SCALES GIVEN
         X0=X1
         Y0=Y1
         TREF=T1
         XREF=XP1
      END IF
      WRITE (*,'(20X,A,F7.4,A,F7.4,A)')
     .' X-SCALE=',XSCL,' KM/IN  ',1./XSCL,' IN/KM'
      WRITE (*,'(20X,A,F7.4,A,F7.4,A,F5.2,A)')
     .' T-SCALE=',TSCL,' S/IN   ',1./TSCL,' IN/S    VRED=',
     .VRED,' KM/S'
  260 PRINT *,' '
```

```
      PRINT *,' ACCEPT SCALES WITH BUTTON "1"'
      PRINT *,' OR SELECT SCALE RESET WITH BUTTON "2":'
      PRINT *,' OR SELECT PROGRAM RESET WITH BUTTON "3":'
      CALL DIGIT (BUTTON,X1,Y1)
      IF (BUTTON.EQ.'1') THEN
         GO TO 300
      ELSE IF (BUTTON.EQ.'2') THEN
         GO TO 70
      ELSE IF (BUTTON.EQ.'3') THEN
         WRITE (*,'(A,F6.3,A)')' VSRC= ',VSRC,
     .   ' KM/S       (VELOCITY NEAR SOURCE)'
          WRITE (*,'(A,F6.3,A)')' VRCV= ',VRCV,
     .   ' KM/S       (VELOCITY NEAR RECEIVER)'
         WRITE (*,'(A,F6.3,A)')' VW=   ',VW,
     .   ' KM/S       (VELOCITY OF WATER COLUMN)'
         WRITE (*,'(A)')' VSRC='
         READ (*,*) VSRC
         WRITE (*,'(A)')' VRCV='
         READ (*,*) VRCV
         WRITE (*,'(A)')' VW=  '
         READ (*,*) VW
         GO TO 40
      END IF
      GO TO 260
C
C MAIN DIGITIZING LOOP OVER LAYERS
C
C GET THE VELOCITY OF THE LAYER JUST BELOW THE CURRENT LAYER
  300 WRITE (*,'(/A,I2)') ' >>>>LAYER=',K1
      PRINT *,' DIGITIZE TWO PTS (SLOPE) OF ARRIVAL WITH BUTTON "1"'
      PRINT *,' OR SELECT KEYBOARD ENTRY OF VELOCITY WITH BUTTON "2"'
      PRINT *,' OR SELECT KEYBOARD LAYER ENTRY WITH BUTTON "3"'
      PRINT *,' OR SELECT NEXT MENU WITH BUTTON "4":'
      CALL DIGIT (BUTTON,X1,Y1)
      IF (BUTTON.EQ.'4') THEN
         PRINT *,' '
         PRINT *,'   SELECT FILE INPUT OF MODEL LAYERS WITH BUTTON "1"'
         PRINT *,'   OR LIST EXISTING MODEL LAYERS WITH BUTTON "2"'
         PRINT *,'   OR SELECT RETRY CURRENT LAYER WITH BUTTON "3"'
         PRINT *,'   OR SELECT NEXT MENU WITH BUTTON "4":'
         CALL DIGIT (BUTTON,XDUM,YDUM)
         IF (BUTTON.EQ."4") THEN
            PRINT *,' '
            PRINT *,'     SELECT SCALE RESET WITH BUTTON "1"'
            PRINT *,'     OR TERMINATE PROGRAM WITH BUTTON "4":'
            CALL DIGIT (BUTTON,XDUM,YDUM)
            IF (BUTTON.EQ."4") THEN
               GO TO 500
            ELSE IF (BUTTON.EQ.'1') THEN
               GO TO 70
            END IF
            GO TO 300
         ELSE IF (BUTTON.EQ.'3') THEN
            GO TO 300
         ELSE IF (BUTTON.EQ.'2') THEN
            WRITE (*,'(20X,A)')
     .      ' LAYER    STWTT      V       DZ       TWTT        Z'
            DO 305 IK=1,K
```

```
            WRITE (*,'(22X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     .      IK,TWT(IK),V(IK),H(IK),TWT(IK)+TRCV+TSRC,Z(IK)
305       CONTINUE
          WRITE (*,'(22X,I3,6X,A4,F9.3)') K1,'??? ',V(K1)
          WRITE (*,'(22X,3X,10X,5X,A3)') '???'
        ELSE IF (BUTTON.EQ.'1') THEN
          WRITE (*,'(/A)') ' NAME OF TWT-V INPUT FILE? '
          READ (*,'(A80)') NAME1
          OPEN (UNIT=11,FILE=NAME1,STATUS='OLD',IOSTAT=IOS)
         IF (IOS.NE.0) THEN
           WRITE (*,'(A,I3)') ' *** OPEN FILE ERROR STATUS=',IOS
            GO TO 300
          END IF
          READ (11,*,END=309) TWT(K1)
          WRITE (*,'(/20X,A)')
     .     ' LAYER    STWTT      V       DZ       TWTT        Z'
306       READ (11,*,END=308) TWT(K1+1),V(K1+1)
            K=K+1
            K1=K1+1
            TWT(K)=TWT(K)-TRCV-TSRC
            H(K)=V(K)*(TWT(K)-TWT(K-1))*0.5
            CALL DELTAU (V(K),H(K),NRAY,P,DTAU,NDT)
            DO 307 I=1,NDT
              TAUSUM(I)=DTAU(I)+TAUSUM(I)
307         CONTINUE
            Z(K)=Z(K-1)+H(K)
            WRITE (10,'(1X,F10.3,F10.3,F10.3,F10.3,F10.3)')
     .      TWT(K)+TRCV+TSRC,V(K),Z(K),H(K),TWT(K)
            WRITE (*,'(22X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     .      K,TWT(K),V(K),H(K),TWT(K)+TRCV+TSRC,Z(K)
          GO TO 306
308       WRITE (*,'(22X,I3,F10.3,F9.3,8X,F9.3)')
     .    K1,TWT(K1),V(K1),TWT(K1)+TRCV+TSRC
309       CLOSE (UNIT=11)
        END IF
        GO TO 300
      ELSE IF (BUTTON.EQ.'1') THEN
        CALL DIGIT (BUTTON,X2,Y2)
        K=K+1
        K1=K1+1
        XD2=(X2-X0)*XSCL + XREF
        TD2=(Y0-Y2)*TSCL + TREF
        DXQ=(X2-X1)*XSCL
        DTQ=(Y1-Y2)*TSCL
        IF (ABS(DXQ).LT..0001) DXQ=SIGN(.0001,DXQ)
        UK1=DTQ/DXQ + PRED
        V(K1)=1./UK1
        WRITE (*,'(/20X,A,F7.4,A)') 'DIGITIZED VELOCITY=',V(K1),' KM/S'
        GO TO 330
      ELSE IF (BUTTON.EQ.'2') THEN
        K=K+1
        K1=K1+1
310     WRITE (*,'(/A)') ' TYPE IN THE LAYER VELOCITY: '
        READ (*,*) V(K1)
        IF (V(K1).LE.0.) THEN
          WRITE (*,'(/A)') ' *** VELOCITY MUST BE .GT. ZERO.'
          GO TO 310
        END IF
```

```
         UK1=1./V(K1)
         GO TO 330
      ELSE IF (BUTTON.EQ.'3') THEN
         K=K+1
         K1=K1+1
 315     WRITE (*,'(/A)') ' TYPE IN VERTICAL TWTT AND LAYER VELOCITY: '
         READ (*,*) TWT(K),V(K1)
         TWT(K)=TWT(K)-TRCV-TSRC
         IF (TWT(K).LT.TWT(K-1)) THEN
            WRITE (*,'(/A,F7.4)') ' *** TWTT MUST BE .GT. ',TWT(K-1)
            GO TO 315
         ELSE IF (V(K1).LE.0.) THEN
            WRITE (*,'(/A)') ' *** VELOCITY MUST BE .GT. ZERO.'
            GO TO 315
         END IF
         UK1=1./V(K1)
         H(K)=V(K)*(TWT(K)-TWT(K-1))*0.5
         CALL DELTAU (V(K),H(K),NRAY,P,DTAU,NDT)
         DO 320 I=1,NDT
            TAU(I)=DTAU(I)+TAUSUM(I)
 320     CONTINUE
         GO TO 470
      END IF
      GO TO 300
C
C GET THE ESTIMATED VERTICAL TWO-WAY TRAVEL TIME TO THE BOTTOM OF THE
C CURRENT LAYER
 330  PRINT *,' '
      PRINT *,
     .' DIGITIZE ESTIMATED VERTICAL TWO-WAY TRAVEL TIME WITH BUTTON "1"'
      PRINT *,
     .' OR SELECT KEYBOARD ENTRY OF TRAVEL TIME ESTIMATE WITH',
     .' BUTTON "2"'
      PRINT *,' OR SELECT NEXT MENU WITH BUTTON "4":'
      CALL DIGIT (BUTTON,XDUM,Y3)
      IF (BUTTON.EQ.'4') THEN
         PRINT *,' '
         PRINT *,'   LIST EXISTING MODEL LAYERS WITH BUTTON "2"'
         PRINT *,'   OR SELECT RETRY CURRENT LAYER WITH BUTTON "3"'
         PRINT *,'   OR TERMINATE PROGRAM WITH BUTTON "4":'
         CALL DIGIT (BUTTON,XDUM,YDUM)
         IF (BUTTON.EQ."4") THEN
            K=K-1
            K1=K1-1
            GO TO 500
         ELSE IF (BUTTON.EQ.'3') THEN
            K=K-1
            K1=K1-1
            GO TO 300
         ELSE IF (BUTTON.EQ.'2') THEN
            WRITE (*,'(/20X,A)')
     .      ' LAYER    STWTT      V       DZ       TWTT        Z'
            DO 335 IK=1,K-1
               WRITE (*,'(22X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     .         IK,TWT(IK),V(IK),H(IK),TWT(IK)+TRCV+TSRC,Z(IK)
 335        CONTINUE
            WRITE (*,'(22X,I3,6X,A4,F9.3)') K,'??? ',V(K)
            WRITE (*,'(22X,3X,10X,F9.3)') V(K1)
```

```
         END IF
         GO TO 330
      ELSE IF (BUTTON.EQ.'1') THEN
         TWT(K)=(Y0-Y3)*TSCL + TREF
         WRITE (*,'(/20X,A,F7.4,A)')
     .   'DIGITIZED ESTIMATE VERTICAL TWTT=',TWT(K),' S'
         GO TO 350
      ELSE IF (BUTTON.EQ.'2') THEN
 340     WRITE (*,'(/A)')
     .   ' TYPE IN VERTICAL TWO-WAY TRAVEL TIME ESTIMATE: '
         READ (*,*) TWT(K)
C
C CORRECT TRAVEL TIME DOWN TO BURIED SOURCE AND RECEIVER (WE PRESUME TRAVEL
C TIME TAKEN FROM ANOTHER SOURCE SUCH AS A SINGLE-CHANNEL SEISMIC SECTION)
         TWT(K)=TWT(K)-TRCV-TSRC
         IF (TWT(K).LT.TWT(K-1)) THEN
            WRITE (*,'(/A,F7.4)') ' *** TWTT MUST BE .GT. ',TWT(K-1)
            GO TO 340
         END IF
         GO TO 350
      END IF
      GO TO 330
C
C GET THE T-X POINT TO BE FIT WHICH LIES EITHER ON THE ARRIVAL PRECRITICALLY
C REFLECTED OFF OF THE BOTTOM OF THE CURRENT LAYER OR ON THE REFRACTED
C ARRIVAL (HEAD WAVE) FROM THE LAYER JUST BELOW.
 350  PRINT *,' '
      PRINT *,' DIGITIZE X,T DATA POINT TO BE MODELED WITH BUTTON "1"'
      PRINT *,' OR SELECT NEXT MENU WITH BUTTON "4":'
      CALL DIGIT (BUTTON,X1,Y1)
      IF (BUTTON.EQ.'4') THEN
         PRINT *,' '
         PRINT *,'    LIST EXISTING MODEL LAYERS WITH BUTTON "2"'
         PRINT *,'    OR SELECT RETRY CURRENT LAYER WITH BUTTON "3"'
         PRINT *,'    OR TERMINATE PROGRAM WITH BUTTON "4":'
         CALL DIGIT (BUTTON,XDUM,YDUM)
         IF (BUTTON.EQ."4") THEN
            K=K-1
            K1=K1-1
            GO TO 500
         ELSE IF (BUTTON.EQ.'3') THEN
            K=K-1
            K1=K1-1
            GO TO 300
         ELSE IF (BUTTON.EQ.'2') THEN
            WRITE (*,'(/20X,A)')
     .      ' LAYER    STWTT      V      DZ      TWTT       Z'
            DO 355 IK=1,K-1
               WRITE (*,'(22X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     .         IK,TWT(IK),V(IK),H(IK),TWT(IK)+TRCV+TSRC,Z(IK)
 355        CONTINUE
            WRITE (*,'(22X,I3,F10.3,A1,F8.3)') K,TWT(K),'?',V(K)
            WRITE (*,'(22X,3X,10X,F9.3)') V(K1)
         END IF
         GO TO 350
      ELSE IF (BUTTON.EQ.'1') THEN
         XD=(X1-X0)*XSCL + XREF
         TD=(Y0-Y1)*TSCL + TREF
```

```
         WRITE (*,'(/20X,A,F7.3,F8.3,A,F5.2,A)')
     .   'DIGITIZED X,T DATUM=',XD,TD,'    (VRED=',VRED,' KM/S)'
         GO TO 400
      END IF
      GO TO 350
C
C EARTH MODEL ITERATION LOOP
  400 ISTEP=0
      WRITE (*,'(/20X,A)')' STEP   STWTT  Tdata   Tmodel   Tdiff'
C
C COMPUTE DELAY TIME CONTRIBUTION OF LAYER
  410 H(K)=V(K)*(TWT(K)-TWT(K-1))*0.5
      CALL DELTAU (V(K),H(K),NRAY,P,DTAU,NDT)
      DO 415 I=1,NDT
         TAU(I)=DTAU(I)+TAUSUM(I)
  415 CONTINUE
C
C COMPUTE TRAVEL TIMES OF PRECRITICAL REFLECTION OFF BOTTOM OF LAYER
C OUT TO POINT AT WHICH REFRACTED ARRIVAL EMERGES FROM LAYER BELOW
      X(1)=0.0
      T(1)=TAU(1)
      TAU(NDT+1)=TAU(NDT) + (TAU(NDT)-TAU(NDT-1))
      DO 420 I=2,NDT
         IF (P(I)*V(K1).GT.1.) GO TO 425
         X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
         T(I)=(P(I)-PRED)*X(I) + TAU(I)
  420 CONTINUE
  425 NDTR=I-1
C
C COMPUTE TRAVEL TIMES OF REFRACTED ARRIVAL FROM LAYER BELOW OUT TO X=XMAX
  430 X(NDTR+1)=X(NDTR) + 1.
      T(NDTR+1)=T(NDTR) + (X(NDTR+1)-X(NDTR))*(UK1-PRED)
      X(NDTR+2)=AMAX1(XMAX,X(NDTR)+2.)
      T(NDTR+2)=T(NDTR) + (X(NDTR+2)-X(NDTR))*(UK1-PRED)
C
C DETERMINE RAY SEGMENTS WHICH BRACKET X=XD
      I1=0
      I2=0
      DO 435 I=1,NDTR+1
          IF (XD.GE.X(I).AND.XD.LE.X(I+1)) THEN
             I1=I
             I2=I+1
             GO TO 450
          END IF
  435 CONTINUE
      WRITE (*,'(/A,F10.4/A)')
     . ' *** RAY TRACED ARRIVAL ONLY EXTENDS OUT TO X=',X(NDTR+2),
     . '      (XMAX IS TOO SMALL).'
  440 PRINT *,' '
      PRINT *,' SELECT REDIGITIZE CURRENT LAYER WITH BUTTON "1"'
      PRINT *,
     .' OR SELECT CHANGE XMAX AND REITERATE WITH BUTTON "2":'
      CALL DIGIT (BUTTON,X1,Y1)
      IF (BUTTON.EQ.'1') THEN
         K=K-1
         K1=K1-1
         GO TO 300
      ELSE IF (BUTTON.EQ.'2') THEN
```

```
        WRITE (*,'(/A,F5.1,A)')' XMAX= ',XMAX,
     .  ' KM         (MAX X OF RAYTRACE)'
  445   WRITE (*,'(/A)')'   XMAX='
        READ (*,*) XMAX
        IF (XMAX.LE.0.0) THEN
           WRITE (*,'(/A)') ' *** XMAX MUST BE .GT. ZERO.'
           GO TO 440
        END IF
        GO TO 400
      END IF
      GO TO 440
C
C COMPUTE SYNTHETIC TRAVEL TIME AT X=XD
  450 STD=(T(I2)-T(I1))*(XD-X(I1))/(X(I2)-X(I1)) + T(I1)
C
C OUTPUT RESULTS OF THIS ITERATION
      ISTEP=ISTEP+1
      WRITE (*,'(20X,I4,F9.3,2F8.3,F10.4)') ISTEP,TWT(K),TD,STD,TD-STD
C
C CHECK FOR CONVERGENCE:
      IF (ABS(TD-STD).GT.DELT) THEN
C
C DID NOT CONVERGE--PERTURB VERTICAL TWO-WAY TRAVEL TIME AND REPEAT
        IF (ISTEP.LT.NSTEP) THEN
           TWT(K)=TWT(K) + (TD-STD)
           GO TO 410
        ELSE
           WRITE (*,'(/A)')' *** VELOCITY ESTIMATES NOT CONVERGING.'
  455      PRINT *,' '
           PRINT *,' SELECT REDIGITIZE CURRENT LAYER WITH BUTTON "1"'
           PRINT *,
     .     ' OR SELECT CHANGE PARAMETERS AND REITERATE WITH BUTTON "2":'
           CALL DIGIT (BUTTON,X1,Y1)
           IF (BUTTON.EQ.'1') THEN
              K=K-1
              K1=K1-1
              GO TO 300
           ELSE IF (BUTTON.EQ.'2') THEN
              WRITE (*,'(/A,F5.1,A)')' XMAX= ',XMAX,
     .        ' KM         (MAX X OF RAYTRACE)'
              WRITE (*,'(A,F5.4,A)')' DELT= ',DELT,
     .        ' S          (MAX TIME MISFIT ERROR)'
              WRITE (*,'(A,I3,A)')  ' NSTEP=',NSTEP,
     .        '              (MAX NUMBER OF ITERATIONS)'
  460         WRITE (*,'(/A)')'   XMAX='
              READ (*,*) XMAX
              WRITE (*,'(A)')'   DELT='
              READ (*,*) DELT
              WRITE (*,'(A)')'   NSTEP='
              READ (*,*)    NSTEP
              IF (XMAX.LE.0.0 .OR. DELT.LE.0.0 .OR. NSTEP.LE.0) THEN
                 WRITE (*,'(/A)')
     .           ' *** XMAX, DELT AND NSTEP MUST BE .GT. ZERO.'
                 GO TO 460
              END IF
              GO TO 400
           END IF
           GO TO 455
```

```
         END IF
      END IF
C
C CONVERGED--OUTPUT END RESULT
  470 Z(K)=Z(K-1)+H(K)
      WRITE (*,'(/A)')' LAYER    STWTT      V      DZ       TWTT       Z'
      WRITE (*,'(2X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     . K,TWT(K),V(K),H(K),TWT(K)+TRCV+TSRC,Z(K)
      WRITE (*,'(2X,I3,10X,F9.3)') K1,V(K1)
C
C LET USER MAKE FINAL DECISION ON PROPRIETY OF SOLUTION
  475 PRINT *,' '
      PRINT *,' ACCEPT THIS LAYER SOLUTION WITH BUTTON "2"'
      PRINT *,' OR SELECT REDIGITIZE CURRENT LAYER WITH BUTTON "3"'
      PRINT *,' OR LIST EXISTING MODEL LAYERS WITH BUTTON "4":'
      CALL DIGIT (BUTTON,X1,Y1)
      IF (BUTTON.EQ.'2') THEN
C
C ACCEPTABLE--WRITE RESULT TO OUTPUT FILE AND UPDATE DELAY TIME INTEGRAL
         WRITE (10,'(1X,F10.3,F10.3,F10.3,F10.3,F10.3)')
     .   TWT(K)+TRCV+TSRC,V(K),Z(K),H(K),TWT(K)
         DO 480 I=1,NDT
            TAUSUM(I)=TAU(I)
  480    CONTINUE
         GO TO 300
      ELSE IF (BUTTON.EQ.'3') THEN
C
C NOT ACCEPTABLE--LET USER RE-DIGITIZE LAYER
         K=K-1
         K1=K1-1
         GO TO 300
C
C USER WAFFLING--LET HIM SEE WHAT HAS TRANSPIRED BEFORE
      ELSE IF (BUTTON.EQ.'4') THEN
         WRITE (*,'(/20X,A)')
     .   ' LAYER    STWTT      V      DZ      TWTT      Z'
         DO 490 IK=1,K-1
            WRITE (*,'(22X,I3,F10.3,F9.3,F8.3,F9.3,F9.3)')
     .      IK,TWT(IK),V(IK),H(IK),TWT(IK)+TRCV+TSRC,Z(IK)
  490    CONTINUE
            WRITE (*,'(22X,I3,A1,F9.3,F9.3,F8.3,F9.3,F9.3)')
     .      K,'?',TWT(K),V(K),H(K),TWT(K)+TRCV+TSRC,Z(K)
         WRITE (*,'(22X,3X,10X,F9.3)') V(K1)
      END IF
      GO TO 475
C
C TERMINATE SESSION
C
C ADD A ONE HALF SECOND THICK BOTTOM LAYER TO PRESERVE SELECTED
C VELOCITY AT BASE OF MODEL
  500 IF (K.LT.1) GO TO 600
      K=K+1
      TWT(K)=TWT(K-1)+0.5
      H(K)=V(K)*0.5
      Z(K)=Z(K-1)+H(K)
      WRITE (10,'(1X,F10.3,F10.3,F10.3,F10.3,F10.3)')
     .   TWT(K)+TRCV+TSRC,V(K),Z(K),H(K),TWT(K)
C
```

```
C CLOSE FILE AND DIGITIZER
  600 CLOSE(UNIT=10)
      CALL DIGEND
      STOP
      END
C-------------------------------------------
      SUBROUTINE DELTAU (V,H,NP,P,DTAU,NDT)
C...COMPUTES THE DELAY TIME CONTRIBUTION DTAU OF A CONSTANT VELOCITY LAYER
C...  V,H     VELOCITY AND THICKNESS OF LAYER
C...  NP      NUMBER OF RAYS TO TRACE
C...  P       ARRAY OF NP RAY PARAMETERS
C...  DTAU    RETURNED ARRAY OF DELAY TIME CONTRIBUTION AT EACH P(I)
C...  NDT     NUMBER OF RAYS SUCCESSFULLY TRACED (IE., PRECRITICAL)
C...
C...COPYRIGHT 1988 BY KEVIN MACKENZIE, UNIV OF TEXAS INSTITUTE FOR GEOPHYSICS
C...
C...THIS PROGRAM AND ITS SUBROUTINES ARE FREEWARE AND MAY BE COPIED,
C...DISTRIBUTED AND USED FREELY, BUT CANNOT BE SOLD OR MODIFIED WITHOUT
C...EXPLICIT WRITTEN CONSENT FROM THE AUTHOR.
C...THE AUTHOR MAKES NO WARRANTIES OR GUARANTEES EXPRESS OR IMPLIED
C...AS TO THE ACCURACY OF THIS CODE OR AS TO THE CONSEQUENCES OF ITS USE.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      REAL*4 DTAU(1),P(1)
      DTAU0=2.*H/V
      VSQ=V*V
      DO 10 I=1,NP
         PSQ=P(I)*P(I)
         IF (PSQ*VSQ.GT.1.) GO TO 20
         DTAU(I)=DTAU0*SQRT(1.-PSQ*VSQ)
   10 CONTINUE
   20 NDT=I-1
      RETURN
      END
C-------------------------------------------
      SUBROUTINE DIGINI (NDEV)
C...SUBROUTINE TO SETUP SERIAL I/O CHANNEL AND INITIALIZE THE ATTACHED
C...DIGITIZER OF THE TYPE INDICATED BY <NDEV>:
C...     =1, IMPLIES A CALCOMP 9600 DIGITIZER (DEFAULT)
C...     =2, IMPLIES A SUMMAGRAPHICS MICROGRID 36X48 INCH DIGITIZER
C...     =3, IMPLIES A SUMMAGRAPHICS SUMMASKETCH PROFESSIONAL MM 12X18
C...             INCH DIGITIZER
C...THIS VERSION PRESUMES THAT SERIAL PORT COMMUNICATIONS (BAUD AND PARITY
C...OF HOST SERIAL PORT AND DIGITIZER) ARE PROPERLY ESTABLISHED BEFORE
C...PROGRAM EXECUTION.
C...
C...COPYRIGHT 1988 BY KEVIN MACKENZIE, UNIV OF TEXAS INSTITUTE FOR GEOPHYSICS
C...
C...THIS PROGRAM AND ITS SUBROUTINES ARE FREEWARE AND MAY BE COPIED,
C...DISTRIBUTED AND USED FREELY, BUT CANNOT BE SOLD OR MODIFIED WITHOUT
C...EXPLICIT WRITTEN CONSENT FROM THE AUTHOR.
C...THE AUTHOR MAKES NO WARRANTIES OR GUARANTEES EXPRESS OR IMPLIED
C...AS TO THE ACCURACY OF THIS CODE OR AS TO THE CONSEQUENCES OF ITS USE.
C...
C $$$$$ FORTRAN 77
```

```
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      CHARACTER*32 COM
      INTEGER*4 NC, NDEV, IDEV, LDEV
      REAL*4 XR0, YR0, CSR, SNR
      CHARACTER*1 CR, LF, ESC
      COMMON /DIGCOM/ IDEV, XR0, YR0, CSR, SNR
      SAVE /DIGCOM/
      SAVE LDEV
      DATA LDEV/0/
C SET DEVICE TYPE FLAG
      IF (NDEV.EQ.0) THEN
         IDEV=1
      ELSE
         IDEV=NDEV
      END IF
      LDEV=IDEV
C INITIALIZE THE SELECTED DIGITIZER
      IF (IDEV.EQ.1) THEN
C ...CALCOMP 9600
         COM=' '
         NC=0
      ELSE IF (IDEV.EQ.2) THEN
C ...SUMMAGRAPHIC MICROGRID 36X48
         CR=CHAR(13)
         ESC=CHAR(27)
C     SET ACQUISITION MODE: '<ESC>M1'= POINT
C     SET SAMPLING RESOLUTION: '<ESC>C2'= 1000 lines per inch
         COM=ESC//'M1'//CR//ESC//'C2'//CR
         NC=8
      ELSE IF (IDEV.EQ.3) THEN
C ...SUMMAGRAPHIC SUMMASKETCH PRO MM 12X18
C     SET ACQUISITION MODE: 'B'= POINT, 'A'= SWITCH STREAM (100 pairs/sec)
C                           '@'= CONTINUOUS STREAM, 'I'= INCREMENT,
C                           'D'= TRIGGER ('P'=TRIGGER CHARACTER)
C     SET SAMPLING RESOLUTION: 'j'= 1000 lines per inch, 'h'= 500 LPI
         CR=CHAR(13)
         COM='B'//CR//'j'//CR
         NC=4
      END IF
      IF (NC.GT.0) WRITE (*,'(A$)') COM(1:NC)
C INITIALIZE ROTATION PARAMETERS
      XR0=0.
      YR0=0.
      CSR=1.
      SNR=0.
      RETURN
      END
C-------------------------------------------
      SUBROUTINE DIGROT (X1,Y1,X2,Y2)
C...DETERMINES ROTATION PARAMETERS TO CORRECT FOR THE TILT IN THE USER'S
C...HORIZONTAL AXIS DESCRIBED BY POINTS <X1,Y1> AND <X2,Y2> (RETURNED BY
C...TWO CALLS TO SUBROUTINE <DIGIT>) RELATIVE TO THE ABSOLUTE HORIZONTAL
C...AXIS OF THE DIGITIZER'S FRAME OF REFERENCE.  THE ROTATION PARAMETERS
C...ARE PLACED IN COMMON BLOCK <DIGCOM> FOR ACCESS BY SUBSEQUENT CALLS TO
C...SUBROUTINE <DIGIT>.  MULTIPLE CALLS TO THIS SUBROUTINE WILL YIELD
C...CORRECT RESULTS AS IT WILL ACCOMODATE PREVIOUS SOFTWARE ROTATIONS.
```

```
C...THIS ROUTINE IS DEVICE INDEPENDENT AND COMPATIBLE WITH ANY HARDWARE
C...ROTATIONS ALREADY IN EFFECT.  NOTE THAT NUMERICAL PRECISION OF
C...DIGITIZED OUTPUT COULD BE AFFECTED IF ROTATION ANGLES EXCEED A FEW
C...DEGREES.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      INTEGER*4 IDEV
      REAL*4 XR0, YR0, CSR, SNR
      COMMON /DIGCOM/ IDEV, XR0, YR0, CSR, SNR
      SAVE /DIGCOM/
C REMOVE ANY PREVIOUS SOFTWARE ROTATION APPLIED TO THE DATA
      SQ=CSR*CSR + SNR*SNR
      XA=(CSR*X1-SNR*Y1)/SQ + XR0
      YA=(SNR*X1+CSR*Y1)/SQ + YR0
      XB=(CSR*X2-SNR*Y2)/SQ + XR0
      YB=(SNR*X2+CSR*Y2)/SQ + YR0
C COMPUTE NEW ROTATION PARAMETERS
      XR=XB-XA
      YR=YB-YA
      HR=SQRT(XR*XR+YR*YR)
      CSR=XR/HR
      SNR=YR/HR
      XR0=XA
      YR0=YA
      RETURN
      END
C-------------------------------------------
      SUBROUTINE DIGIT (BUTTON,X,Y)
C...GETS A DIGITIZED POINT, RETURNING CHARACTER OF BUTTON
C...PUSHED AND CURSOR LOCATION X,Y IN REAL INCHES FROM ABSOLUTE ORIGIN
C...OF DIGITIZING TABLET.  IF ROUTINE DIGROT CALLED PREVIOUSLY,
C...RETURNED COORDINATES ARE ROTATED TO HORIZONTAL AS DEFINED BY DIGROT AND
C...THE VALUES OF THE COORDINATES ARE RELATIVE TO THE FIRST POINT OF THE
C...HORIZONTAL REFERENCE LINE.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      CHARACTER*256 COM
      CHARACTER*1 BUTTON
      INTEGER*4 IDEV
      REAL*4 X, Y, DX, DY, XR0, YR0, CSR, SNR
      COMMON /DIGCOM/ IDEV, XR0, YR0, CSR, SNR
      SAVE /DIGCOM/
C
C  IDEV=1
C  CALCOMP 9600 DIGITIZER (SET FOR ASCII FORMAT #3) OUTPUTS A 13-BYTE
C  ASCII STRING:  'KMIXXXXIYYYY<CR>'  WHERE IXXXX,IYYYY=ABSOLUTE X AND Y
C  LOCATIONS IN 10**-3 INCHES, K=CHARACTER VALUE OF BUTTON PUSHED ON
C  DIGITIZING PUCK (0-9,*,#,A,B,C,D) AND M=PEN STATUS (IE, U=UP, D=DOWN).
C
C  IDEV=2
C  SUMMAGRAPHICS MICROGRID 36x48 GRAPHICS TABLET OUTPUTS 19-BYTE
C  STRING IN 1000 COUNTS PER INCH ASCII MODE: 'SIXXXX,SIYYYY,KK,T<CR>'.
```

```
C  NOTE EXPLICIT COMMAS.  IXXXX,IYYYY=ABSOLUTE X AND Y LOCATIONS IN 10**-3
C  INCHES. S=SIGN OF COORDINATE (+ OR -) AND KK=TWO CHARACTER VALUE OF
C  BUTTON PUSHED ON 4-BUTTON DIGITIZING PUCK (01,02,03,04) OR STYLUS
C  (01=STYLUS TIP, 02=STYLUS BARREL) OR 16-BUTTON DIGITIZING PUCK (01-16).
C  IF NO BUTTON  PUSHED (E.G., INCREMENT MODE), KK=00.  T=TABLET AREA
C  IDENTIFIER AND IS ALWAYS SET TO ZERO BY DIGITIZER.
C
C  IDEV=3
C  SUMMAGRAPHICS SUMMASKETCH PRO MM 12x18 GRAPHICS TABLET OUTPUTS 15-BYTE
C  STRING IN 1000 COUNTS PER INCH ASCII MODE: 'IXXXX,IYYYY,K<CR><LF>'.
C  NOTE EXPLICIT COMMAS.  IXXXX,IYYYY=ABSOLUTE X AND Y LOCATIONS IN 10**-3
C  INCHES AND K=CHARACTER VALUE OF BUTTON PUSHED ON DIGITIZING PUCK
C  (1,2,3,4) OR STYLUS (5=STYLUS TIP, 6=STYLUS BUTTON, 7=BOTH TOGETHER).
C  IN 500 COUNTS PER INCH RESOLUTION MODE, FORMAT IS  'IXXX,IYYY,K<CR><LF>'
C  (13-BYTE STRING) WHERE IXXX,IYYY=ABSOLUTE X AND Y LOCATIONS IN
C  0.5*10**-2 INCHES (4-DIGIT RESOLUTION).
C
      IF (IDEV.EQ.1) THEN
C ...CALCOMP 9600
         READ (*,'(A1,1X,2F5.3,1X)') BUTTON,X,Y
      ELSE IF (IDEV.EQ.2) THEN
C ...SUMMAGRAPHICS MICROGRID 36X48
C    NOTE THAT WE ONLY READ SECOND CHARACTER OF BUTTON ARGUMENT (IE., 0-9)
         READ (*,'(F6.3,1X,F6.3,2X,A1,3X)') X,Y,BUTTON
      ELSE IF (IDEV.EQ.3) THEN
C ...SUMMAGRAPHICS SUMMASKETCH PRO MM 12X18
         READ (*,'(F5.3,1X,F5.3,1X,A1,2X)') X,Y,BUTTON
      END IF
C PERFORM AXIAL ROTATION (NOTE THAT THIS WILL NOT EFFECT DATA IF
C ROUTINE DIGROT NEVER CALLED)
      DX=X-XR0
      DY=Y-YR0
      X=CSR*DX+SNR*DY
      Y=-SNR*DX+CSR*DY
      RETURN
      END
C-----------------------------------------
      SUBROUTINE DIGEND
C...SUBROUTINE TO TERMINATE SERIAL CHANNEL TO DIGITIZER.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      CHARACTER*256 COM
      INTEGER*4 IDEV
      REAL*4 XR0, YR0, CSR, SNR
      COMMON /DIGCOM/ IDEV, XR0, YR0, CSR, SNR
      SAVE /DIGCOM/
C CLOSE SERIAL PORT TO DIGITIZER
      COM=' '
      IDEV=0
      XR0=0.
      YR0=0.
      CSR=1.
      SNR=0.
      RETURN
      END
```

APPENDIX II -- Ray-Tracing Software

```
      PROGRAM PTXT
C
C...COMPUTES P-TAU AND X-T CURVES FROM AN INPUT TWO-WAY TRAVEL TIME AND
C...VELOCITY MODEL ASSUMING CONSTANT VELOCITY FLAT LAYERS.  LOW VELOCITY
C...ZONES IN THE MODEL ARE ACCEPTABLE.  WILL GENERATE PRECRITICAL
C...REFLECTIONS, REFRACTIONS (HEAD WAVES), WATER COLUMN MULTIPLES (OF
C...REFLECTIONS AND REFRACTIONS) AND SURFACE GHOST REFLECTIONS (GENERATED
C...NEAR RECEIVER) (OF REFLECTIONS AND REFRACTIONS).  ASSUMES SOURCE AND
C...RECEIVER BURIED IN FIRST LAYER (WATER COLUMN) AND ALLOWS FOR SEPARATE
C...LAYER 1, NEAR-SOURCE AND NEAR-RECEIVER VELOCITIES TO BETTER REPRESENT
C...DIRECT WATER WAVES IN DEEP WATER ENVIRONMENTS. IF XMAX.GT.0, REFRACTIONS
C...ARE COMPUTED.  IF MULT.GT.0, MULTIPLES ARE COMPUTED. IF NGHOST.GT.0,
C...SURFACE GHOST REFLECTIONS (SURFACE REFLECTIONS NEAR BURIED RECEIVER) ARE
C...COMPUTED.  ZRCV AND/OR ZSRC MAY BE ZERO HOWEVER VSRC AND VRCV MUST BE
C...GREATER THAN ZERO.  IF VRED.GT.0, VELOCITY REDUCTION IS APPLIED TO ALL
C...OUTPUT TRAVEL TIMES.  TWO OUTPUT FILES ARE GENERATED CONTAINING ALL
C...REQUESTED PHASES FOR EVERY LAYER IN THE MODEL V(K),TWT(K):
C...     P(I,J,K),TAU(I,J,K)    (P(I,,).LT.P(I+1,,))        (FORMAT=2F10.4)
C...     X(I,J,K),T(I,J,K)      (X(I,,).LT.X(I+1,,))        (FORMAT=2F10.4).
C...EACH VALUE OF I REPRESENTS A RAY OF THE JTH PHASE INTO THE KTH LAYER.
C...THAT IS ALL RAYS FOR A GIVEN PHASE (E.G. PRIMARY REFLECTION) FOR A
C...GIVEN LAYER ARE FOLLOWED BY THE NEXT PHASE (E.G. 1ST MULTIPLE
C...REFLECTION) FOR THE SAME LAYER AND SO ON UNTIL ALL REQUESTED PHASES
C...FOR THAT LAYER ARE OUTPUT.  THEN ALL RAYS FOR EACH PHASE IN THE NEXT
C...LAYER ARE OUTPUT.  RAYS FOR EACH PHASE FOR EACH LAYER AND EACH LAYER
C...ARE SEPARATED IN BOTH FILES BY DUMMY POINTS= (9999.,9999.) (WHICH CAN
C...BE EXPLOITED TO CAUSE A PEN-UP MOVE BETWEEN CURVES WHEN PLOTTED).
C...FOR A GIVEN PHASE IN EITHER FILE, I WILL BE .GE.2  AND .LE.NRAY.
C...ALL PHASES REQUESTED WILL BE OUTPUT IN THE FOLLOWING ORDER AFTER THE
C...DIRECT WATER WAVE (A PRIMARY REFRACTION):
C...   PRIMARY REFLECTION,
C...   SURFACE GHOST OF PRIMARY REFLECTION,
C...   1ST AND SUBSEQUENT MULTIPLE REFLECTIONS,
C...   PRIMARY REFRACTION,
C...   SURFACE GHOST OF REFRACTION,
C...   1ST AND SUBSEQUENT MULTIPLES OF REFRACTION.
C...NOTE THAT IF ONE OR MORE LOW VELOCITY ZONES ARE PRESENT, NOT ALL PHASES
C...WILL BE PRESENT AS DICTATED BY THE PHYSICS (E.G., REFRACTIONS FROM THE
C...LVZ).
C...
C...COPYRIGHT 1988 BY KEVIN MACKENZIE, UNIV OF TEXAS INSTITUTE FOR GEOPHYSICS
C...
C...THIS PROGRAM AND ITS SUBROUTINES ARE FREEWARE AND MAY BE COPIED,
C...DISTRIBUTED AND USED FREELY, BUT CANNOT BE SOLD OR MODIFIED WITHOUT
C...EXPLICIT WRITTEN CONSENT FROM THE AUTHOR.
C...THE AUTHOR MAKES NO WARRANTIES OR GUARANTEES EXPRESS OR IMPLIED
C...AS TO THE ACCURACY OF THIS CODE OR AS TO THE CONSEQUENCES OF ITS USE.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS ROUTINE: DELTAU
C...
      CHARACTER NAME*80,ANS*1
      REAL*4 V(0:20),TWT(0:20)
      REAL*4 P(201),T(200),X(200)
      REAL*4 DTAU(200),TAU(201),TAUSUM(201),WTAU(201),GTAU(201)
```

```
      XDUM=9999.0
      V(0)=0.
      TWT(0)=0.
      ZSRC=.000       !DEPTH OF SOURCE .122=400FT  .018=60FT
      ZRCV=.018       !DEPTH OF RECEIVER
      VSRC=1.495      !VELOCITY NEAR SOURCE
      VRCV=1.495      !VELOCITY NEAR RECEIVER
      VRED=6.         !REDUCING VELOCITY OF SEISMIC PLOT (VRED=0.0=INFINITY)
      XMAX=50.        !MAXIMUM REFRACTION OFFSET
      NRAY=100        !MAXIMUM NUMBER OF RAYS TRACED
      MULT=0          !NUMBER OF WATER COLUMN MULTIPLES
      NGHOST=0        !ENABLE SEASURFACE GHOST REFLECTION
C
      WRITE (*,'(40X,A)')     'PTXT 3.0'
      WRITE (*,'(40X,A///)') 'Copyright 1988 by K. MacKenzie'
      WRITE (*,'(A)') ' NAME OF TWT-V INPUT FILE? '
      READ (*,'(A80)') NAME
      OPEN (UNIT=10,FILE=NAME,STATUS='OLD')
      WRITE (*,'(A)') ' NAME OF X-T REFLECTION OUTPUT FILE? '
      READ (*,'(A80)') NAME
      OPEN (UNIT=11,FILE=NAME,STATUS='NEW')
C      WRITE (*,'(A)') ' NAME OF P-TAU OUTPUT FILE? '
      WRITE (*,'(A)') ' NAME OF X-T REFRACTION OUTPUT FILE? '
      READ (*,'(A80)') NAME
      OPEN (UNIT=12,FILE=NAME,STATUS='NEW')
C
C READ IN MODEL
      NK=0
      DO 10 K=1,100
         READ (10,*,END=15) TWT(K),V(K)
   10 CONTINUE
   15 NK=K-1
      CLOSE (UNIT=10)
      V(NK+1)=999.
   20 WRITE (*,'(/A,F8.4,A)')' ZSRC=  ',ZSRC,
     .' KM         (SOURCE DEPTH)'
      WRITE (*,'(/A,F8.4,A)')' ZRCV=  ',ZRCV,
     .' KM         (RECEIVER DEPTH)'
      WRITE (*,'(A,F7.3,A)')' VSRC=  ',VSRC,
     .' KM/S       (VELOCITY NEAR SOURCE)'
      WRITE (*,'(A,F7.3,A)')' VRCV=  ',VRCV,
     .' KM/S       (VELOCITY NEAR RECEIVER)'
      WRITE (*,'(A,F7.3,A)')' VRED=  ',VRED,
     .' KM/S       (REDUCING VELOCITY OF PLOT,VRED=0.0=INFINITY)'
      WRITE (*,'(A,F7.3,A)')' XMAX=  ',XMAX,
     .' KM          (MAX X OF RAYTRACE)'
      WRITE (*,'(A,I3,A)')' NRAY=  ',NRAY,
     .'                (NUMBER OF RAYS TRACED)'
      WRITE (*,'(A,I3,A)')' MULT=  ',MULT,
     .'                (NUMBER OF SEAFLOOR MULTIPLES)'
      WRITE (*,'(A,I3,A)')' NGHOST=',NGHOST,
     .'                (ENABLE SEASURFACE GHOST: 1=YES, 0=NO)'
      WRITE (*,'(/A)')
     .   ' CHANGE ZSRC,ZRCV,VSRC,VRCV,VRED,XMAX,NRAY,MULT,NGHOST? (N) '
      READ (*,'(A1)') ANS
      IF (ANS.EQ.'Y' .OR. ANS.EQ.'y') THEN
         WRITE (*,'(/A)')'   ZSRC='
         READ (*,*) ZSRC
```

```
         WRITE (*,'(A)')'   ZRCV='
         READ (*,*) ZRCV
         WRITE (*,'(A)')'   VSRC='
         READ (*,*) VSRC
         WRITE (*,'(A)')'   VRCV='
         READ (*,*) VRCV
         WRITE (*,'(A)')'   VRED='
         READ (*,*) VRED
         IF (VRED.GT.50.) VRED=0.
         IF (VRED.LE.0.) VRED=0.
         WRITE (*,'(A)')'   XMAX='
         READ (*,*) XMAX
         WRITE (*,'(A)')'   NRAY='
         READ (*,*) NRAY
         WRITE (*,'(A)')'   MULT='
         READ (*,*) MULT
         WRITE (*,'(A)')'   NGHOST='
         READ (*,*) NGHOST
         IF (ZRCV.LE.0. .AND. NGHOST.GT.0) NGHOST=0
         GO TO 20
      END IF
      IF (NRAY.LT.2 .OR. NRAY.GT.200) THEN
         WRITE (*,'(/A/)')' *** NRAY MUST BE .GE. 2 AND .LE. 200'
         GO TO 20
      END IF
      IF (VSRC.LE.0. .OR. VRCV.LE.0. .OR. VW.LE.0.) THEN
         WRITE (*,'(/A)')' *** VELOCITIES MUST BE .GT. ZERO.'
         GO TO 20
      END IF
      IF (ZSRC.LT.0.) ZSRC=0.
      IF (ZRCV.LT.0.) ZRCV=0.
      USRC=1./VSRC
      URCV=1./VRCV
      UW=1./V(1)
      IF (VRED.NE.0.) THEN
         PRED=1./VRED
      ELSE
         PRED=0.
      END IF
      UWR=AMAX1(USRC,URCV,UW)
C
C SET UP LIST OF RAYS (P=0.0 TO P=UWR) AND ZERO OUT TAUSUM
      DO 30 I=1,NRAY+1
         TAUSUM(I)=0.0
         TAU(I)=0.0
         WTAU(I)=0.0
         GTAU(I)=0.0
         P(I)=UWR*REAL(I-1)/REAL(NRAY-1)
   30 CONTINUE
C
C PRECORRECT TAUSUM FOR SOURCE BURIED IN FIRST LAYER
      IF (ZSRC.GT.0.) THEN
         HSRC=ZSRC*0.5
         CALL DELTAU (VSRC,HSRC,NRAY,P,TAU,NDT)
         DO 40 I=1,NDT
            TAUSUM(I)=-TAU(I)
   40    CONTINUE
      END IF
```

```
      TSRC=ZSRC*UW
C
C PRECORRECT TAUSUM FOR RECEIVER BURIED IN FIRST LAYER
      IF (ZRCV.GT.0.) THEN
         HRCV=ZRCV*0.5
         CALL DELTAU (VRCV,HRCV,NRAY,P,TAU,NDT)
         DO 50 I=1,NDT
            TAUSUM(I)=TAUSUM(I)-TAU(I)
   50    CONTINUE
C
C PRECOMPUTE SURFACE GHOST CONTRIBUTION
         IF (NGHOST.GT.0) THEN
            CALL DELTAU (VRCV,ZRCV,NRAY,P,GTAU,NDT)
            GTAU(NDT+1)=GTAU(NDT) + (GTAU(NDT)-GTAU(NDT-1))
         END IF
      END IF
      TRCV=ZRCV*UW
C
C COMPUTE DIRECT WATER WAVE TRAVEL TIME TABLE ACCOUNTING FOR ANY DIFFERENCE
C IN SOURCE AND RECEIVER DEPTHS
      ZDIF=ABS(ZRCV-ZSRC)
      VMEAN=(VRCV+VSRC)*0.5
      UMEAN=1./VMEAN
      IF (ZDIF.GT.0.) THEN
         HDIF=ZDIF*0.5
         CALL DELTAU (VMEAN,HDIF,NRAY,P,TAU,NDT)
         X(1)=0.0
         T(1)=TAU(1)
         TAU(NDT+1)=TAU(NDT) + (TAU(NDT)-TAU(NDT-1))
         DO 60 I=2,NDT
            X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
            T(I)=(P(I)-PRED)*X(I) + TAU(I)
   60    CONTINUE
C
C OUTPUT DIRECT WATER WAVE AS REFRACTION FOR FIRST LAYER
         WRITE (12,'(2F10.4)') (X(I),T(I),I=1,NDT)
      ELSE
         NDT=1
         X(NDT)=0.0
         T(NDT)=0.0
      END IF
C
C EXTEND DIRECT WAVE OUT TO REFRACTION DISTANCES
      IF (XMAX.GT.X(NDT)) THEN
         DELX=XMAX-X(NDT)
         DX=DELX/(.20*NRAY)
         IX=NINT(DELX/DX) + 1
         DO 70 I=1,IX
            XINC=REAL(I-1)*DX
            X(I)=X(NDT) + XINC
            T(I)=T(NDT) + XINC*(UMEAN-PRED)
   70       CONTINUE
         WRITE (12,'(2F10.4)') (X(I),T(I),I=1,IX)
      END IF
      WRITE (12,'(2F10.4)') XDUM,XDUM
C
C PRECOMPUTE MULTIPLE CONTRIBUTION
      IF (MULT.GT.0) THEN
```

```
         H=V(1)*TWT(1)*0.5
         CALL DELTAU (V(1),H,NRAY,P,WTAU,NDT)
         WTAU(NDT+1)=WTAU(NDT) + (WTAU(NDT)-WTAU(NDT-1))
      END IF
      NDTOLD=NRAY
C
C LOOP OVER LAYERS COMPUTING ALL REQUESTED PHASES
      DO 200 K=1,NK
         U=1./V(K)
         U1=1./V(K+1)
         LVZ=0
         H=V(K)*(TWT(K)-TWT(K-1))*0.5
         CALL DELTAU (V(K),H,NRAY,P,DTAU,NDT)
         IF (NDT.GT.NDTOLD) THEN
            LVZ=1
            NDT=NDTOLD
         END IF
         NDTOLD=NDT
         DO 80 I=1,NDT
            TAU(I)=DTAU(I)+TAUSUM(I)
   80    CONTINUE
C
C PRECRITICAL REFLECTIONS
C    ...PRIMARIES
         X(1)=0.0
         T(1)=TAU(1)
         TAU(NDT+1)=TAU(NDT)+(TAU(NDT)-TAU(NDT-1))
         DO 90 I=2,NDT
            X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
            T(I)=(P(I)-PRED)*X(I) + TAU(I)
   90    CONTINUE
         WRITE (11,'(2F10.4)') (X(I),T(I),I=1,NDT)
         WRITE (11,'(2F10.4)') XDUM,XDUM
C         WRITE (12,'(2F10.4)') (P(I),TAU(I),I=1,NDT)
C         WRITE (12,'(2F10.4)') XDUM,XDUM
         INEW=U1/UWR*(NRAY-1) + 1
         XNEW=X(INEW)
         TNEW=T(INEW)
         DO 100 I=1,NDT+1
            TAUSUM(I)=TAU(I)
  100    CONTINUE
C    ...GHOST REFLECTION
         IF (NGHOST.GT.0) THEN
            DO 110 I=1,NDT+1
               TAU(I)=GTAU(I)+TAUSUM(I)
  110       CONTINUE
            X(1)=0.0
            T(1)=TAU(1)
            DO 120 I=2,NDT
               X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
               T(I)=(P(I)-PRED)*X(I) + TAU(I)
  120       CONTINUE
            WRITE (11,'(2F10.4)') (X(I),T(I),I=1,NDT)
            WRITE (11,'(2F10.4)') XDUM,XDUM
C            WRITE (12,'(2F10.4)') (P(I),TAU(I),I=1,NDT)
C            WRITE (12,'(2F10.4)') XDUM,XDUM
         END IF
C    ...MULTIPLE REFLECTIONS
```

```
        IF (MULT.GT.0) THEN
           DO 150 M=1,MULT
              DO 130 I=1,NDT+1
                 TAU(I)=M*WTAU(I)+TAUSUM(I)
  130         CONTINUE
              X(1)=0.0
              T(1)=TAU(1)
              DO 140 I=2,NDT
                 X(I)=-(TAU(I+1)-TAU(I-1))/(P(I+1)-P(I-1))
                 T(I)=(P(I)-PRED)*X(I) + TAU(I)
  140         CONTINUE
              WRITE (11,'(2F10.4)') (X(I),T(I),I=1,NDT)
              WRITE (11,'(2F10.4)') XDUM,XDUM
C              WRITE (12,'(2F10.4)') (P(I),TAU(I),I=1,NDT)
C              WRITE (12,'(2F10.4)') XDUM,XDUM
  150      CONTINUE
        END IF
C
C REFRACTIONS
C    ...PRIMARIES
        IF (XMAX.GT.0.0 .AND. K.GT.1 .AND. LVZ.EQ.0) THEN
           DELX=XMAX-XOLD
           DX=DELX/(.20*NRAY)
           IX=NINT(DELX/DX) + 1
           DO 160 I=1,IX
              XINC=REAL(I-1)*DX
              X(I)=XOLD + XINC
              T(I)=TOLD + XINC*(U-PRED)
  160      CONTINUE
           WRITE (12,'(2F10.4)') (X(I),T(I),I=1,IX)
           WRITE (12,'(2F10.4)') XDUM,XDUM
C    ...GHOST REFLECTION
           IF (NGHOST.GT.0) THEN
              GX=-(GTAU(IOLD+1)-GTAU(IOLD-1))/(P(IOLD+1)-P(IOLD-1))
              GT=(P(IOLD)-PRED)*GX + GTAU(IOLD)
              WRITE (12,'(2F10.4)') (X(I)+GX,T(I)+GT,I=1,IX)
              WRITE (12,'(2F10.4)') XDUM,XDUM
           END IF
C    ...MULTIPLE REFLECTIONS
        IF (MULT.GT.0) THEN
           WX=-(WTAU(IOLD+1)-WTAU(IOLD-1))/(P(IOLD+1)-P(IOLD-1))
           WT=(P(IOLD)-PRED)*GX + WTAU(IOLD)
           DO 170 M=1,MULT
              WRITE (12,'(2F10.4)') (X(I)+M*WX,T(I)+M*WT,I=1,IX)
              WRITE (12,'(2F10.4)') XDUM,XDUM
  170      CONTINUE
        END IF
        END IF
        IOLD=INEW
        XOLD=XNEW
        TOLD=TNEW
  200 CONTINUE
C
C EXIT
  300 CLOSE(UNIT=11)
      CLOSE(UNIT=12)
      STOP
      END
```

```
C-------------------------------------------
      SUBROUTINE DELTAU (V,H,NP,P,DTAU,NDT)
C...COMPUTES THE DELAY TIME CONTRIBUTION DTAU OF A CONSTANT VELOCITY LAYER
C...  V,H     VELOCITY AND THICKNESS OF LAYER
C...  NP      NUMBER OF RAYS TO TRACE
C...  P       ARRAY OF NP RAY PARAMETERS
C...  DTAU    RETURNED ARRAY OF DELAY TIME CONTRIBUTION AT EACH P(I)
C...  NDT     NUMBER OF RAYS SUCCESSFULLY TRACED (IE., PRECRITICAL)
C...
C...COPYRIGHT 1988 BY KEVIN MACKENZIE, UNIV OF TEXAS INSTITUTE FOR GEOPHYSICS
C...
C...THIS PROGRAM AND ITS SUBROUTINES ARE FREEWARE AND MAY BE COPIED,
C...DISTRIBUTED AND USED FREELY, BUT CANNOT BE SOLD OR MODIFIED WITHOUT
C...EXPLICIT WRITTEN CONSENT FROM THE AUTHOR.
C...THE AUTHOR MAKES NO WARRANTIES OR GUARANTEES EXPRESS OR IMPLIED
C...AS TO THE ACCURACY OF THIS CODE OR AS TO THE CONSEQUENCES OF ITS USE.
C...
C $$$$$ FORTRAN 77
C...
C $$$$$ CALLS NO OTHER ROUTINES
C...
      REAL*4 DTAU(1),P(1)
      DTAU0=2.*H/V
      VSQ=V*V
      DO 10 I=1,NP
         PSQ=P(I)*P(I)
         IF (PSQ*VSQ.GT.1.) GO TO 20
         DTAU(I)=DTAU0*SQRT(1.-PSQ*VSQ)
   10 CONTINUE
   20 NDT=I-1
      RETURN
      END
```

APPENDIX III -- Published Abstract

EOS, Trans. AGU, 68, 1374, 1987.

The Shallow Velocity Structure of the Broken Ridge

KEVIN MACKENZIE (Institute for Geophysics, University of Texas at Austin, 8701 Mopac Blvd., Austin, TX 78759-8345)

A high resolution seismic refraction study of the central Broken Ridge was performed in September 1986 in conjunction with the single channel seismic survey conducted by J. Weissel, G. Mountain and G. Karner aboard R/V ROBERT CONRAD. Both the SSQ57A sonobuoys and the seismic streamer outputs were sampled at 1 msec intervals. The seismic source consisted of twin 80 cu. in. water guns fired at a nominal shot spacing of 50 m. More than 50 sonobuoys were deployed along dip and strike lines of the ~3000 km of SCS profiling in the principal 120 x 230 km study area. The sonobuoy data exhibit very good delineation of precritical reflections and wide angle arrivals from the Neogene to Cretaceous sedimentary sequence. Total sub-seafloor penetration approaches 1.5 km although arrivals from basaltic basement are generally very weak if present. There is very good correlation of most of the precritical reflection events with mappable horizons observed in the SCS data. Joint travel time inversions of both the wide-angle and precritical arrivals resolve layer thicknesses on scales of 50-75 m. This is sufficient to map the pinch-out of the lagoonal-like deposits noted by Weissel at the crest of the ridge. The derived structure should enable accurate assessment of depths to proposed drilling targets for the planned ODP sites on the Broken Ridge.