1990-07

An Investigation of Timing Synchronization Errors for Tracking Underwater Vehicles

Read, Robert R.

Monterey, California. Naval Postgraduate School

(2)

NPS55-90-15

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AN INVESTIGATION OF TIMING SYNCHRONIZATION ERRORS FOR TRACKING UNDERWATER VEHICLES

Robert R. Read

July 1990

Prepared for:
Naval Postgraduate School,
Monterey, CA 93955

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

Rear Admiral R. W. West, Jr.	Harrison Shull
Superintendent	Provost

This report was prepared under the joint support of Naval Undersea Warfare Engineering Station, Keyport, Washington and the Naval Postgraduate School Research Program.

This report was prepared by:

R R Read

ROBERT R. READ
Professor of Operations Research

Reviewed by:

P Purdue

PETER PURDUE
Professor and Chairman
Department of Operations Research

Released by:

G E Schacher

Dean of Faculty and Graduate Studies

Unclassified

Security Classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification UNCLASSIFIED	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report
2b Declassification/Downgrading Schedule	Approved for public release; distribution is unlimited
4 Performing Organization Report Number(s) NPS55-90-15	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School; 6b Office Symbol (If Applicable) OR	7a Name of Monitoring Organization NUWES
6c Address (city, state, and ZIP code) Monterey, CA 93943-5000	7b Address (city, state, and ZIP code) Code 512, Keyport, WA 98345
8a Name of Funding/Sponsoring Organization Naval Postgraduate School; 8b Office Symbol (If Applicable) OR/Re	9 Procurement Instrument Identification Number N0002488WX48044AC
8c Address (city, state, and ZIP code) Monterey, California	10 Source of Funding Numbers

Program Element Number	Project No	Task No	Work Unit Accession No

11 Title (Include Security Classification) An Investigation of Timing Synchronization Errors for Tracking Underwater Vehicles

12 Personal Author(s) Read, Robert R.

13a Type of Report	13b Time Covered	14 Date of Report (year, month,day)	15 Page Count
Technical	From To	1990, July	

16 Supplementary Notation The views expressed in this paper are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (continue on reverse if necessary and identify by block number)
Field	Group	Subgroup	Tracking; calibration; systematic errors; components of variance; synchronization modeling.

19 Abstract (continue on reverse if necessary and identify by block number

Two separate sensing arrays track the same underwater vehicle. The two versions of track are different, but roughly parallel. One possible explanation is the presence of a timing synchronization error. The report provides a model for describing and correcting such errors, estimation algorithms for quantifying the model parameters, and statistical methodology for testing the validity of the effects. The techniques are applied to real data, and the results are reported. Also some properties of the noise process are recorded.

20 Distribution/Availability of Abstract [X] unclassified/unlimited [ ] same as report [ ] DTIC users	21 Abstract Security Classification Unclassified
22a Name of Responsible Individual R. R. Read	22b Telephone (Include Area code) (408) 646-2382; 22c Office Symbol OR/Re

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted security classification of this page

All other editions are obsolete Unclassified

AN INVESTIGATION OF TIMING SYNCHRONIZATION ERRORS FOR TRACKING UNDERWATER VEHICLES

R. R. Read

Key Words

Tracking; calibration; systematic errors; components of variance; synchronization modeling

Abstract

Two separate sensing arrays track the same underwater vehicle. The two versions of track are different, but roughly parallel. One possible explanation is the presence of a timing synchronization error. The report provides a model for describing and correcting such errors, estimation algorithms for quantifying the model parameters, and statistical methodology for testing the validity of the effects. The techniques are applied to real data, and the results are reported. Also some properties of the noise process are recorded.

Acknowledgements

This report had the partial support of the Naval Undersea Weapons Engineering Station, Keyport, Washington. The computer programming and graphical work was performed by Colin Cooper. The manuscript was processed by Hania La Born.

1. INTRODUCTION

This report deals with some calibration problems of monitoring a multiple array underwater tracking range. The arrays in the system are of the short baseline type; each contains four sonar transducers placed rigidly at the corners of a cube in a manner that describes a Cartesian coordinate system in three dimensions. Figure 1 contains a diagram showing the structure for these arrays and indicating the numerous signals they may receive. It is slightly deceptive in that real ray paths are not straight lines.

An array receives a distinctive signal from a synchronously timed pinger attached to the target vehicle. The differentials of the sound wavefront's times of arrival at the four hydrophones allow the computation of the azimuth and elevation angles (spherical coordinate longitude and latitude) of the normal to the wavefront at the origin of the local coordinate system. Then, assuming direct path propagation, one can ray trace using Snell's law, [1], starting with the aforementioned elevation angle and utilizing a velocity-versus-depth profile for the speed of sound in the water. Finally, the time differential between the source pulse at the target vehicle and its arrival at the array is used to stop the ray-tracing algorithm and determine the location of the target relative to the array. The local track is the sequential set of these estimated positions.

Each array in the system operates over a limited radius. As the target sojourns through the range, it is tracked by a number of these arrays. See Figure 2 for a plan view of the Nanoose range. (The zero level in the vertical is taken as mean sea level.) The overall path is constructed by transforming each piece of local track to the coordinates of the range based upon the assumed location and orientation of the various local coordinate systems. Discontinuities, or mismatches occur because the track produced by one array does not mesh well with that produced by a neighboring array in the overlap regions. Such mismatches can be seen in Appendix B.

It appears that there are several sources of systematic error in the operation of this system. The question of individual array location and orientation has been treated previously [2]. The present report addresses the timing synchronization problem, from a statistical point of view. That is, the pulses received at the arrays are timed to the range clock with great precision. The timing information must be transmitted to the pinger prior to the release of the target vehicle. Although there is no engineering reason to suspect noticeable error in this transfer, some data exhibit behavior consonant with such an interpretation. Thus we build a mathematical model to account for such a source of systematic error, and develop statistical methodology for interpreting the data in the light of the model. Indeed, if the method provided a way to eliminate mismatches for an entire run for a single vehicle, there would be great temptation to use it as a smoothing filter.

Generally there are several sequences of time points, called point count sets, for which two arrays simultaneously produce track. These occur for tracking in the overlap regions indicated in Figure 2. The paired tracking data of these point counts are called "crossover data." It is the crossover data generated as a result of the target vehicle's entire trip that provide the evidence suggesting timing synchronization problems and the data base for evaluating the use of such a model. Each version of track in a crossover data set is assumed to have been converted to the (common) range coordinate system. The components of this system are called "downrange, crossrange, vertical," or sometimes "centerline, crossline, vertical."

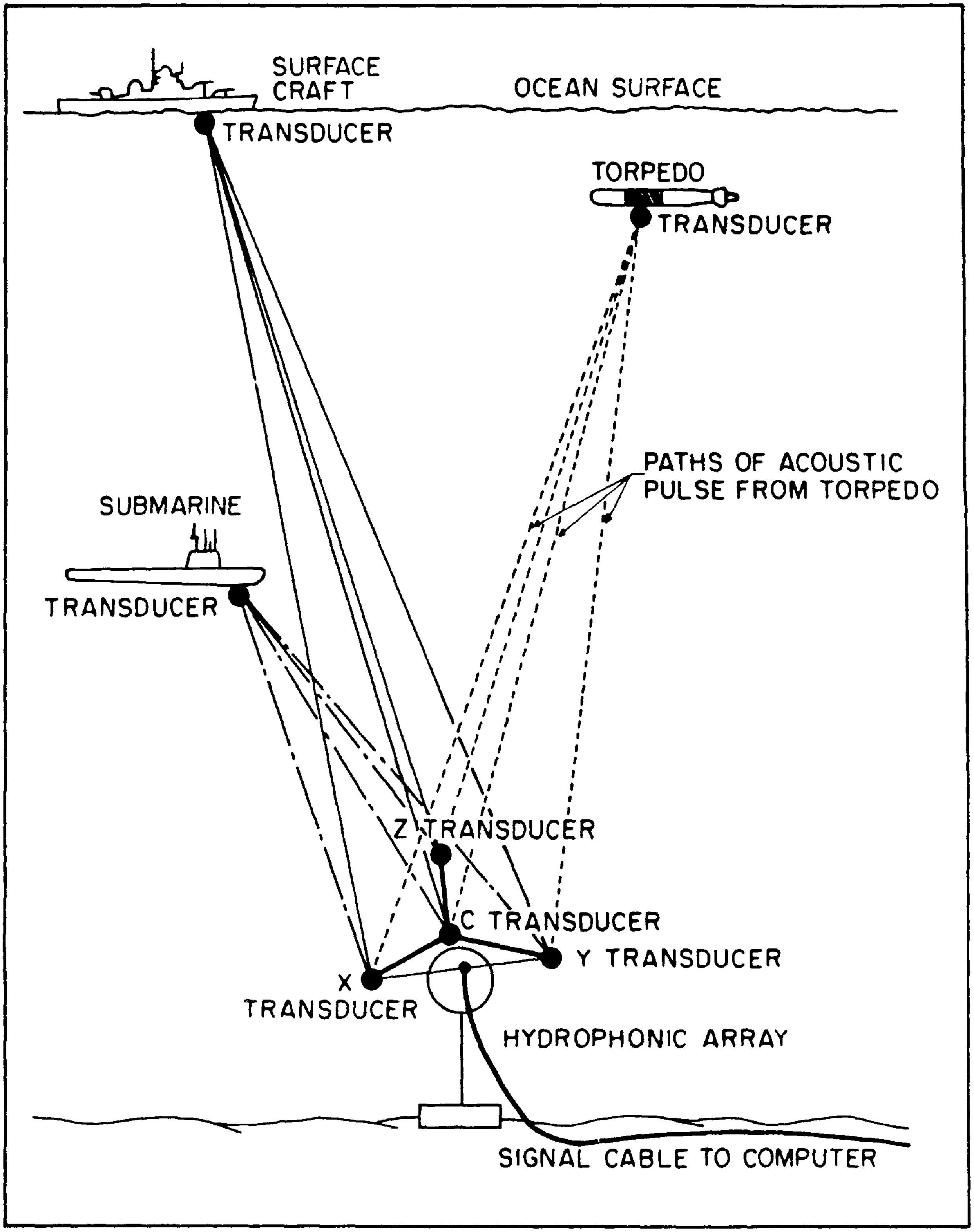


Figure 1. Short Base Line Array and Ray Paths.

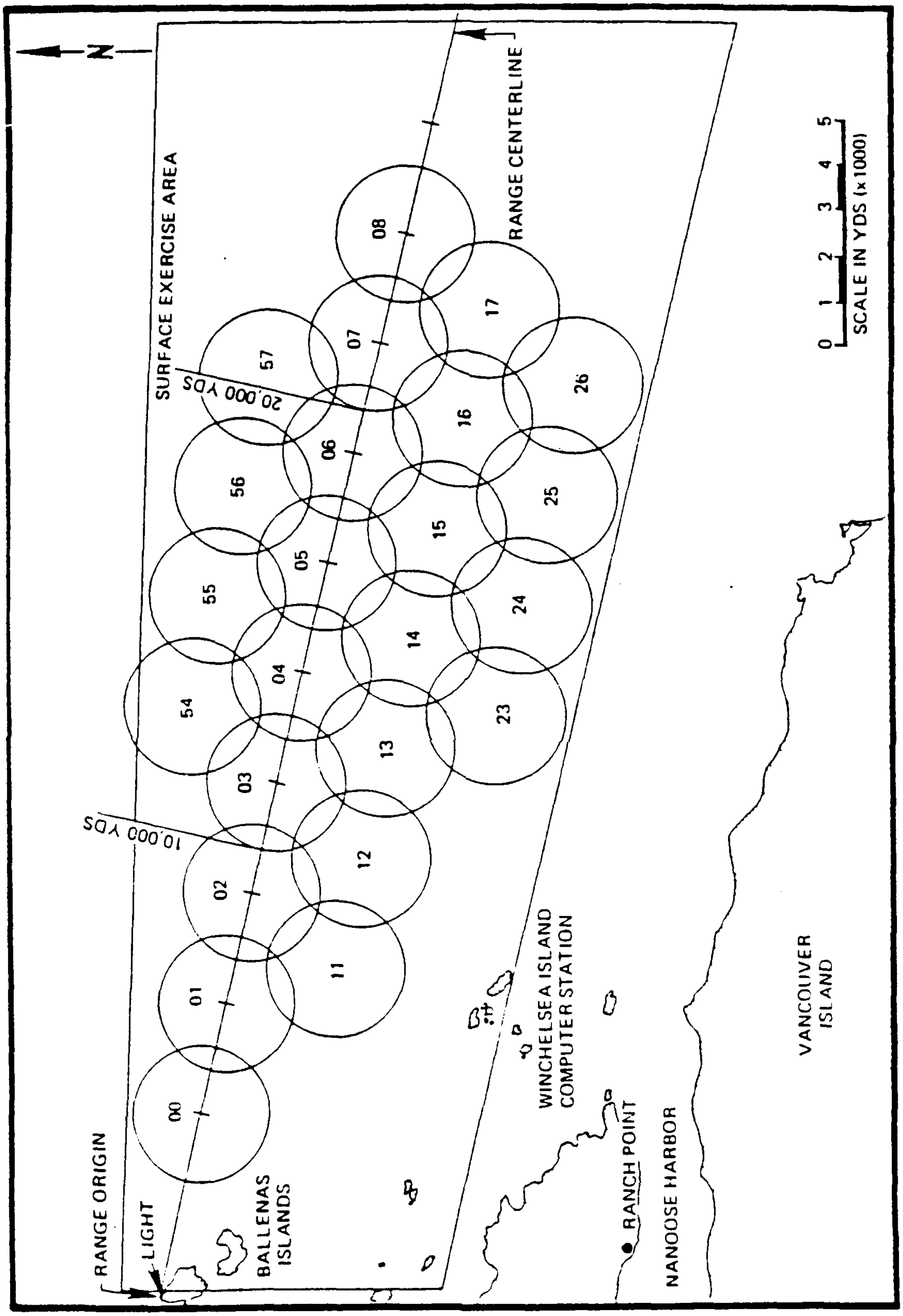


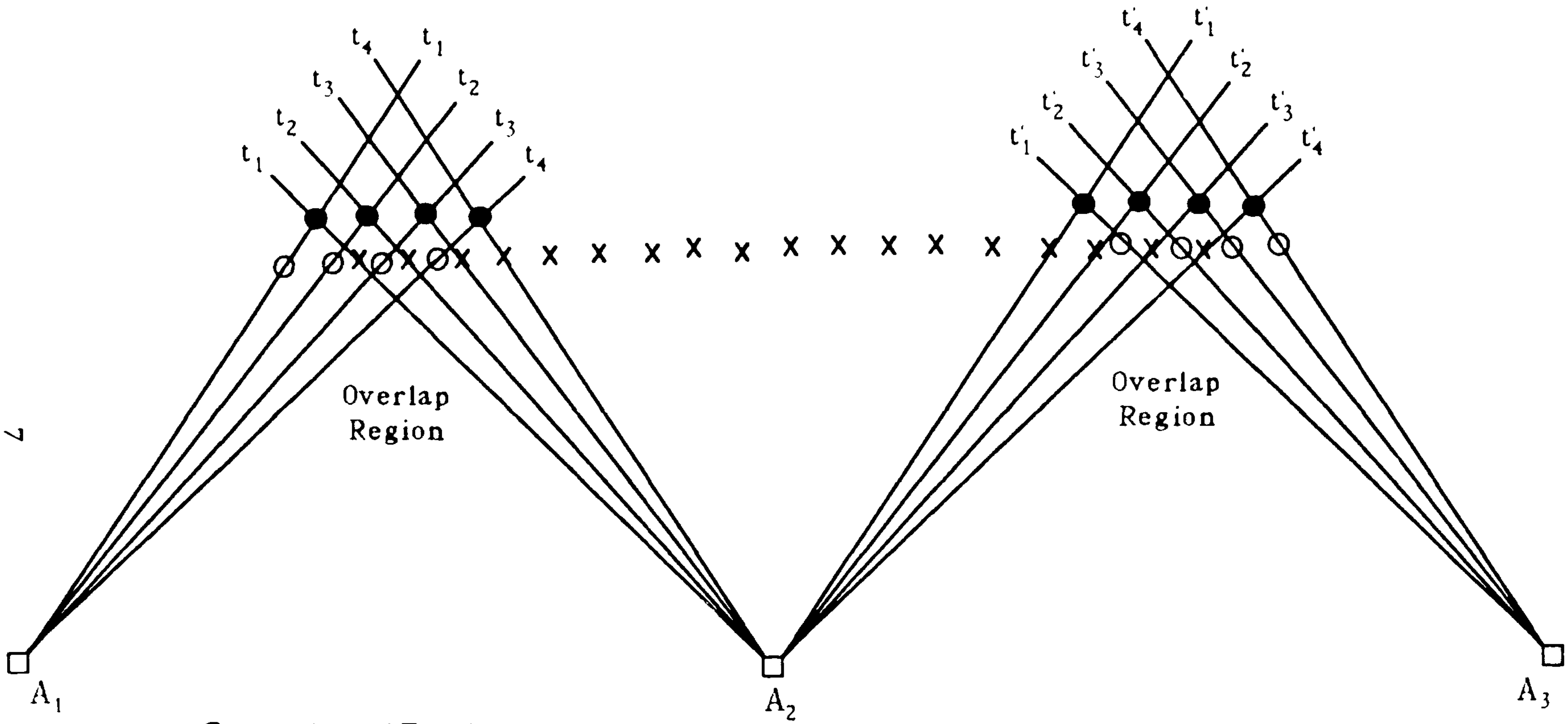
Figure 2. Nanoose Range

The model is developed in Section 2. It allows for both a timing offset and a drift. Estimation methods and statistical properties are developed. Section 3 contains applications to real data. It is shown that the offsets are significant and the drifts are negligible. The issue of the reality of the effects is also treated. I.e., is it the timing offset constant for all crossover regions used in a given run? To answer this question, some modeling of the components of variance is required and when done, a statistical analysis is performed. It appears that the true cause of these effects is something other than a timing synchronization bias. The analyses are supported by some characteristics of the noise process, and these are presented in Section 4. Conclusions are summarized in Section 5. A number of appendices are included that contain supporting data and information, including source codes for computations.

2. MODEL DEVELOPMENT AND STATISTICAL PROPERTIES

Figure 3 contains a mockup illustrating conditions that support the consideration of a timing synchronization correction. It shows a plan view which includes the radial lines from the arrays to the estimated track points in adjacent overlap regions. Between these regions, much data is supplied only by array A_2. The analyst does not see the radial lines on his screen, nor does he see the black dots. He sees only the X's and 0's (no distinction between the two) and no mismatch is apparent. (Mismatches would be apparent however for track pointing in a different direction.) But when one pairs up the radial lines by common time points, then one sees that the two versions of track lack coherence and can be improved by stretching the estimated points to the black dot positions. This can be achieved by a single constant adjustment to the transit time values in the ray tracing algorithm.

Figure 3 : Effect of Timing Offset Errors (Synthesized)



● Actual Track

x Track according to Array A_2

○ Track according to Arrays A_1 or A_3 as indicated

The figure also helps one to imagine the distinction between the effects of an array location error model and a timing adjustment model. If the former were applied to the situation in Figure 3, at least two arrays would have to be moved. But any such decisions cannot be made in isolation. The ultimate goal is the simultaneous improvement of coherence in all overlap areas and, of course, discontinuities must not be created at other places.

A crossover data set is a set of matched pairs of three vectors

$$X(t) = \begin{Bmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \end{Bmatrix} \qquad Y(t) = \begin{Bmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \end{Bmatrix} \qquad t = 1, \ldots, T \tag{1}$$

representing two versions of the same vehicle track for a common set of time values (point counts), T in number. The array that produced $X(t)$ is located at α and the one that produced $Y(t$' is located at β. Thus these two versions of track can be represented in their local coordinate systems as

$$\xi(t) = X(t)-\alpha \qquad \eta(t) = Y(t)-\beta \tag{2}$$

Timing synchronization error corrections may be viewed as either stretching or contracting the vectors $\xi(t)$ and $\eta(t)$ by the same (time) amount. The effect of such corrections is not constant, but depends upon (i) the speed of sound in water at the depths $X_3(t)$ and $Y_3(t)$; (ii) the elevation angles of the ray traces at these times and depths.

The magnitudes of the time adjustments are small and the effects can be described using first order terms

$$\begin{aligned} \xi(t,g(t)) &= \xi(t) + g(t)a(t) \\ \eta(t,g(t)) &= \eta(t) + g(t)b(t) \end{aligned} \tag{3}$$

where $g(t)$ is a scalar function of time representing the total adjustment for timing offset (δ) and drift (m)

$$g(t) = \delta + m(t-t_0) \tag{4}$$

for a conveniently chosen central time, t_0; and a(t), b(t) are the scaled directions of stretch

$$a(t) = v(t)\begin{Bmatrix} \cos(\theta(t))\cos(\phi(t)) \\ \cos(\theta(t))\sin(\phi(t)) \\ \sin(\theta(t)) \end{Bmatrix} \tag{5}$$

i.e., v(t) is the speed of sound at depth $\xi_3(t)$, $\theta(t)$ is the elevation angle of the ray trace from α at $\xi_3(t)$, and $\phi(t)$ is the azimuth of $\xi(t)$ from α. The vector b(t) is defined similarly for the track $\eta(t)$ and its origin β.

It is easily seen that, for given δ, m and t_0, the corrected versions of track in the (common) range coordinate system are

$$\begin{aligned} X(t,g(t)) &= X(t) + g(t)a(t) \\ Y(t,g(t)) &= Y(t) + g(t)b(t) \end{aligned} \tag{6}$$

The statistical estimation problem is to chose δ and m so that the two versions of track agree as well as possible. The least squares approach is adopted. Using the notation

$$\underset{t}{\mathrm{Ave}}\|W(t)\|^2 = \frac{1}{T}\sum_{t=1}^{T} W'(t)W(t) \tag{7}$$

we set up the objective function

$$Q = \underset{t}{\text{Ave}}\left\| X(t,g(t)) - Y(t,g(t)) \right\|^2$$

$$= \underset{t}{\text{Ave}}\left\| X(t) - Y(t) \right\|^2 + \underset{t}{\text{Ave}}[g(t)]^2 \left\| a(t) - b(t) \right\|^2 \tag{8}$$

$$+2\underset{t}{\text{Ave}}\ g(t)[a(t) - b(t)]' [X(t) - Y(t)]$$

and find the values of δ and m that minimize Q. When the two partial derivatives are set equal to zero it is seen that there is convenience in choosing t_0 so that

$$\underset{t}{\text{Ave}}(t - t_0)\left\| a(t) - b(t) \right\|^2 = 0 \tag{9}$$

That is

$$t_0 = \underset{t}{\text{Ave}}\ t\left\| a(t) - b(t) \right\|^2 / \underset{t}{\text{Ave}}\ \left\| a(t) - b(t) \right\|^2$$

This done, the normal equations may be expressed as

$$\begin{aligned} \delta\, \underset{t}{\text{Ave}}\left\| a(t) - b(t) \right\|^2 &= -\underset{t}{\text{Ave}}(a(t) - b(t))' [X(t) - Y(t)] \\ m\, \underset{t}{\text{Ave}}(t - t_0)^2 \left\| a(t) - b(t) \right\|^2 &= -\underset{t}{\text{Ave}}(t - t_0)[a(t) - b(t)]' [X(t) - Y(t)] \end{aligned} \tag{10}$$

and one can readily solve explicitly for the minimizing values $\hat{\delta}$ and $\hat{m}$. Positive values of $\hat{\delta}$ are associated with **stretching**, negative with **contracting**.

If we let $\hat{Q}$ be the minimizing value of Q, it is useful to establish the decomposition

$$Q = \hat{Q} + \left(\hat{\delta} - \delta\right)^2 \underset{t}{\text{Ave}}\left\| a(t) - b(t) \right\|^2 + (\hat{m} - m)^2 \underset{t}{\text{Ave}}(t - t_0)\left\| a(t) - b(t) \right\|^2 \tag{11}$$

Before justifying (11) it is convenient to shorten the writing: let $W(t,\delta(t)) = X(t,g(t)) - Y(t,g(t))$; $c(t) = a(t) - b(t); \hat{g}(t) = \hat{\delta} + \hat{m}(t - t_0)$. Then the statement says

$$\underset{t}{\text{Ave}}\|W(t,g(t))\|^2$$

$$= \underset{t}{\text{Ave}}\|W(t,\hat{g}(t))\|^2 + \left(\hat{\delta} - \delta\right)^2 \underset{t}{\text{Ave}}\|c(t)\|^2 + (\hat{m} - m)^2 \underset{t}{\text{Ave}}(t - t_0)\|c(t)\|^2 \tag{12}$$

and the result is justified by use of the three orthogonality relationships

$$\begin{gathered}
\underset{t}{\text{Ave}}\left[W(t,0) + \hat{\delta}c(t) + \hat{m}(t - t_0)c(t)\right]' c(t) = 0 \\
\underset{t}{\text{Ave}}\left[W(t,0) + \hat{\delta}c(t) + \hat{m}(t - t_0)c(t)\right]' (t - t_0)c(t) = 0 \\
\underset{t}{\text{Ave}}(t - t_0)c'(t)c(t) = 0
\end{gathered} \tag{13}$$

which in turn are established using the normal equations (10), and (9).

The significance of the offset and drift parameters can be judged if we develop the means and variances of $\hat{\delta}$, $\hat{m}$. Letting E denote the mathematical expectation operator, we begin with the assumption

$$E[W(t,g(t)] = 0 \tag{14}$$

which embraces the idea that the two corrected versions of track produce common track without systematic error.

Further, let

$$K_\delta = \left[\sum_t c'(t)c(t)\right]^{-1}; \qquad K_m = \left[\sum_t (t - t_0)^2 c'(t)c(t)\right]^{-1} \tag{15}$$

and use (13) and (14) to show that the estimators $\hat{\delta}$ and $\hat{m}$ are unbiased.

$$E\left(\hat{\delta}\right) = -K_\delta \sum_t c'(t) E[W(t,0)] = K_\delta \sum_t c'(t)[\delta + m(t - t_0)]c(t) = \delta$$

$$E(\hat{m}) = -K_m \sum_t (t - t_0)c'(t) E[W(t,0)] = K_m \sum_t (t - t_0)c'(t)[\delta + m(t - t_o)]c(t) = m$$

To develop variances, we assume that the positive lag covariances are zero; support for this appears in the section on noise characteristics. Let M represent the (zero lag) covariance matrix. (See Appendix C for estimates).

$$M = E[W(t, g(t))W'(t, g(t))] \tag{16}$$

Then one easily represents

$$\mathrm{var}\left(\hat{\delta}\right) = K_\delta^2 \sum_t c'(t) M c(t) \tag{17}$$

$$\mathrm{var}(\hat{m}) = K_m^2 \sum_t (t - t_0)^2 c'(t) M c'(t) \tag{18}$$

$$\mathrm{cov}\left(\hat{\delta}, \hat{m}\right) = K_\delta K_m \sum_t (t - t_0) c'(t) M c(t) \tag{19}$$

If M is proportional to the identity matrix then this last term is zero; the other two terms simplify immensely; and a standard regression development can be used. But the study of noise characteristics does not support this. On the other hand the vectors {c(t)} do not change much with t and this has the tendency to render (19) to be small. The reason for this stability is that crossover data occurs only at the greater distances from the sensing arrays; the azimuth and elevation angles and the layer sound speeds do not change much.

It appears that the matrix M changes with the crossover data set. Methodology for estimating it appears in Section 4. Estimates of the M matrices appear in Appendix C.

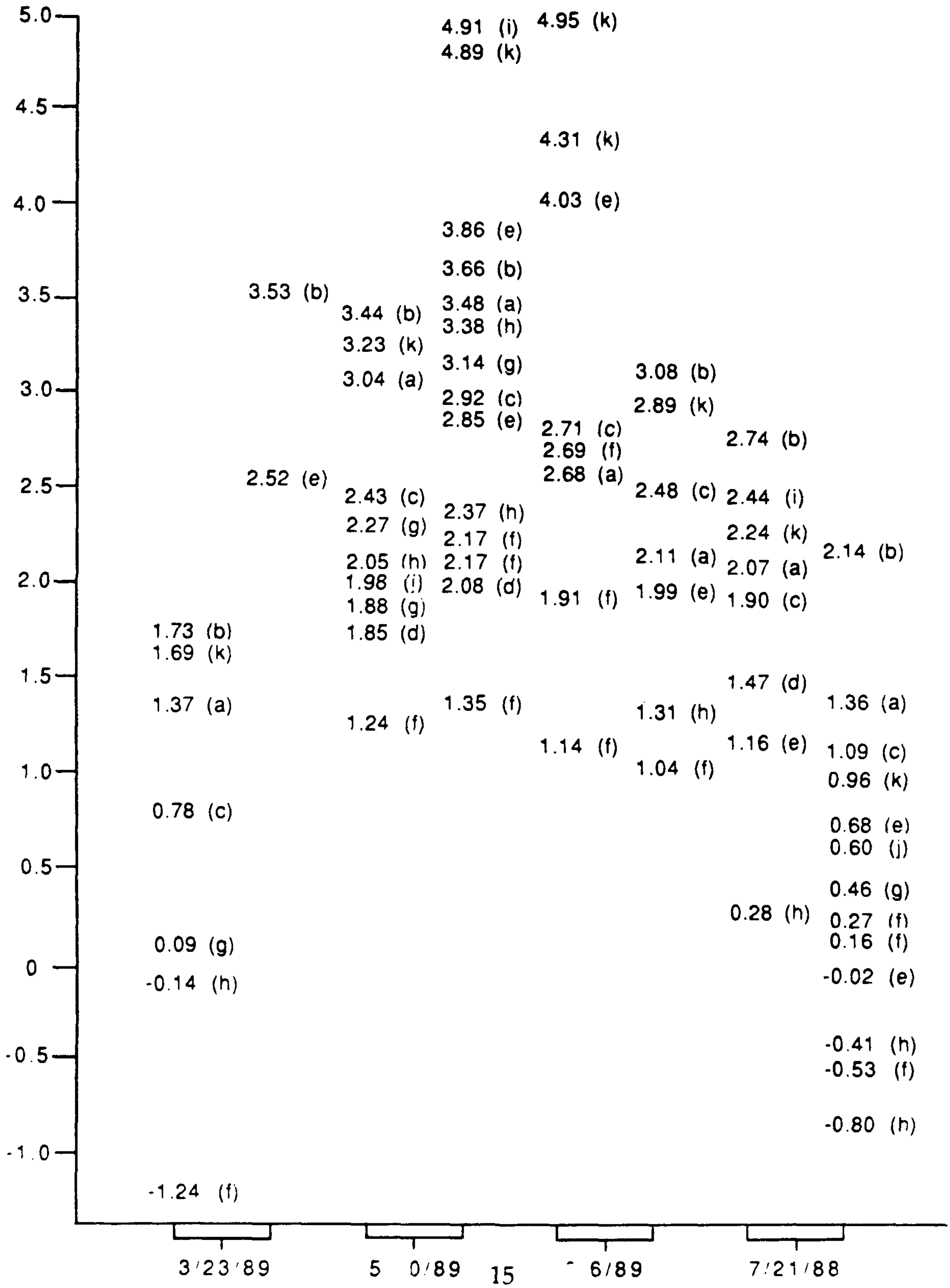
3. DATA ANALYSIS

The data consists of $\hat{\delta}$, $\hat{m}$, and supporting values for 69 segments of crossover track collected over four separate days with two (temporally serial, not concurrent) target vehicles (runs) per day. The information is summarized in Table 1. Missing variance estimates indicate either a data shortage or outlier problems. In a few cases the tracks were curved, and the straight line model is not adequate. The units are milliseconds for $\hat{\delta}$, and milliseconds per point count for $\hat{m}$. One millisecond translates to about 4.5 to 5 feet of distance. The ratios of means to standard deviations are used to judge whether the effects are significantly different from zero. Virtually all of the offsets and some of the drifts are significant although the latter are not strongly so. The "r" column contains the correlations between $\hat{\delta}$ and $\hat{m}$. They are insignificant. A further search for large scale drift was made by plotting $\hat{\delta}$ against the crossover central time t_0 for each run. They appear in Figure 4. If a smooth signal were discernible then we would have a way to connect the δ values that appear in each column of Table 2. But they are chaotic and provide no incentive to continue any concern about drift. It is concluded that the offsets are significant and the drifts are not. The latter will be dropped from further consideration. Some graphical examples of the effect our timing corrections have been selected and appear in Appendix B.

Table 1: Offsets, Drifts, Signal to Noise Ratios

Date	Vehicle	δ	σ_δ	δ/σ_δ	m	σ_m	m/σ_m	r	t_0	T
3/23/89	MK 30	1.37	.0461	29.82	.00168	.001627	1.03	-.00	405.13	42
		1.74	.0431	40.28	-.00831	.002217	3.75	.00	422.72	33
		1.69	.0925	18.29	-.01559	.007024	2.22	.06	4171.09	19
		.78	.0464	16.87	.00245	.001588	1.54	.02	439.71	45
		-.15	.0504	2.97	.01304	.001438	9.07	.02	2012.80	57
		.10	.0802	1.23	.00150	.005509	.27	.01	3080.16	22
		-1.24	.0480	25.96	-.00119	.001645	.72	.01	2531.21	49
3/23/89	MK 30	3.54	.0479	73.85	-.00395	.002119	1.86	.06	422.64	37
		2.52	.0584	43.17	-.00224	.002136	1.05	.11	1605.22	47
5/10/89	MK 27	3.04	.0283	107.40	.00993	.001142	8.70	-.02	3107.58	43
		3.44	.0192	178.88	-.00250	.000866	2.89	-.06	3120.22	38
		3.24	.0401	80.68	-.00007	.002747	.02	.05	6936.70	25
		2.43	.0270	90.18	.00187	.001375	1.36	.00	3116.86	34
		1.86	.0266	69.80	.00029	.000825	.35	-.00	3781.03	56
		1.24	.0228	54.56	.00408	.000630	6.48	-.05	5471.41	59
		2.06	.0197	104.59	.00437	.000662	6.60	.02	4720.56	50
		1.99	.0104	190.86	-.00067	.000364	1.84	.02	8060.02	50
		2.27	.0180	126.16	-.00196	.000770	2.54	-.03	5992.35	40
		1.89	.0868	21.72	-.00001	.000567	.01	-.01	9615.69	50
5/10/89	MK 27	3.49	.0194	180.11	.01062	.000746	14.23	-.03	1733.39	45
		3.66	.0270	135.56	-.00044	.001268	.35	.00	1751.02	37
		4.90	.0189	259.57	.00443	.000689	6.43	.01	10196.44	45
		2.93	.0258	113.27	-.00545	.001358	4.02	.00	1747.85	33
		2.09	.0406	51.50	.00038	.001887	.20	-.01	2366.96	37
		2.85	.0243	117.13	-.00094	.000845	1.11	.07	2980.01	50
		3.87	.----	---.--	-.00050	.------	-.--	.--	12354.91	50
		2.37	.----	---.--	-.01074	.------	-.--	.--	5316.80	15
		3.38	.0131	257.71	.00257	.000458	5.62	.00	12215.15	50
		1.36	.0209	64.92	.00276	.000662	4.17	-.06	3585.00	55
		2.18	.0181	120.55	-.00195	.000628	3.11	.07	4478.00	50
		2.18	.0149	146.48	.00192	.000479	4.02	-.03	5873.92	54
		4.91	.0096	510.80	.00022	.000337	.64	.02	11235.86	50
		3.15	.0281	111.97	-.00202	.001001	2.02	-.05	6480.62	46
6/6/89	MK 27	2.68	.0336	79.78	.01680	.003073	5.47	.02	1952.64	19
		4.31	.0116	371.65	-.00189	.000377	5.02	.08	10562.53	50
		4.95	.0269	184.19	.00144	.001136	1.27	-.04	14411.90	41
		2.72	.0125	217.40	.00343	.000434	7.91	.02	2124.12	50
		4.03	.0136	297.47	.00242	.000464	5.22	-.02	12709.75	50
		1.14	.0376	30.32	-.00197	.001555	1.27	-.01	4979.22	40
		1.91	.0251	76.25	.00401	.001141	3.52	.07	6197.42	35
		2.69	.0151	178.04	-.00143	.000486	2.93	-.05	12044.01	51
6/6/89	MK 27	2.11	.0232	91.30	.00091	.000783	1.17	.03	357.26	50
		3.09	.0434	71.15	.00596	.003137	1.90	-.02	421.96	24
		2.90	.0195	148.86	-.00231	.000677	3.42	-.03	4135.53	50
		2.48	.0233	106.78	-.00211	.000692	3.06	.04	487.61	50
		1.99	.0156	127.36	-.00192	.000541	3.55	-.02	1781.82	50
		1.04	.0202	51.69	-.00075	.000681	1.10	.04	2505.37	50
		1.32	.0181	72.74	.00003	.000631	.05	.01	1991.06	50
7/21/88	MK 27	2.07	.0217	95.45	.00060	.000725	.83	-.02	352.15	52
		2.75	.0300	91.58	.00020	.001845	.11	-.00	377.91	28
		2.24	.0115	195.77	-.00160	.000398	4.01	-.00	4440.01	50
		1.90	.0147	129.27	.00244	.000510	4.79	.03	469.17	50
		1.47	.----	--.--	-.00746	.------	-.--	.--	1062.61	15
		1.16	.0128	90.47	-.00287	.001310	2.19	.01	1772.98	17
		.29	.0144	19.86	.00232	.000501	4.64	-.01	2068.58	50
		2.45	.0112	217.86	.00069	.000392	1.76	.00	3137.66	50
7/21/88	MK 30	1.37	.0245	55.85	.01149	.000853	13.46	.02	1108.15	50
		2.15	.0308	69.77	.00069	.002324	.30	-.02	1152.98	23
		.96	.0192	50.18	.00019	.000662	.29	-.03	6658.19	50
		1.10	.0171	63.97	.00206	.000595	3.46	.02	217.45	50
		.69	.0189	36.48	-.00233	.000656	3.54	-.02	4360.14	50
		-.02	.0099	2.07	.00190	.000346	5.49	-.06	11498.22	50
		-.80	.0209	38.31	.00147	.000730	2.01	-.00	4667.48	50
		-.42	.0337	12.35	.00355	.001173	3.02	-.03	10430.20	50
		.27	.0176	15.46	.00035	.000611	.57	-.07	3793.99	50
		.17	.0251	6.69	.00024	.000952	.26	.03	5141.88	27
		-.54	.0496	10.88	-.00171	.003582	.48	.03	12031.94	24
		.61	.0259	23.51	.00509	.000897	5.67	.02	3011.04	50
		.47	[illegible]	15.94	[illegible]	.001380	2.34	-.04	5697.10	37

Table 2: Estimates of the Timing Offset



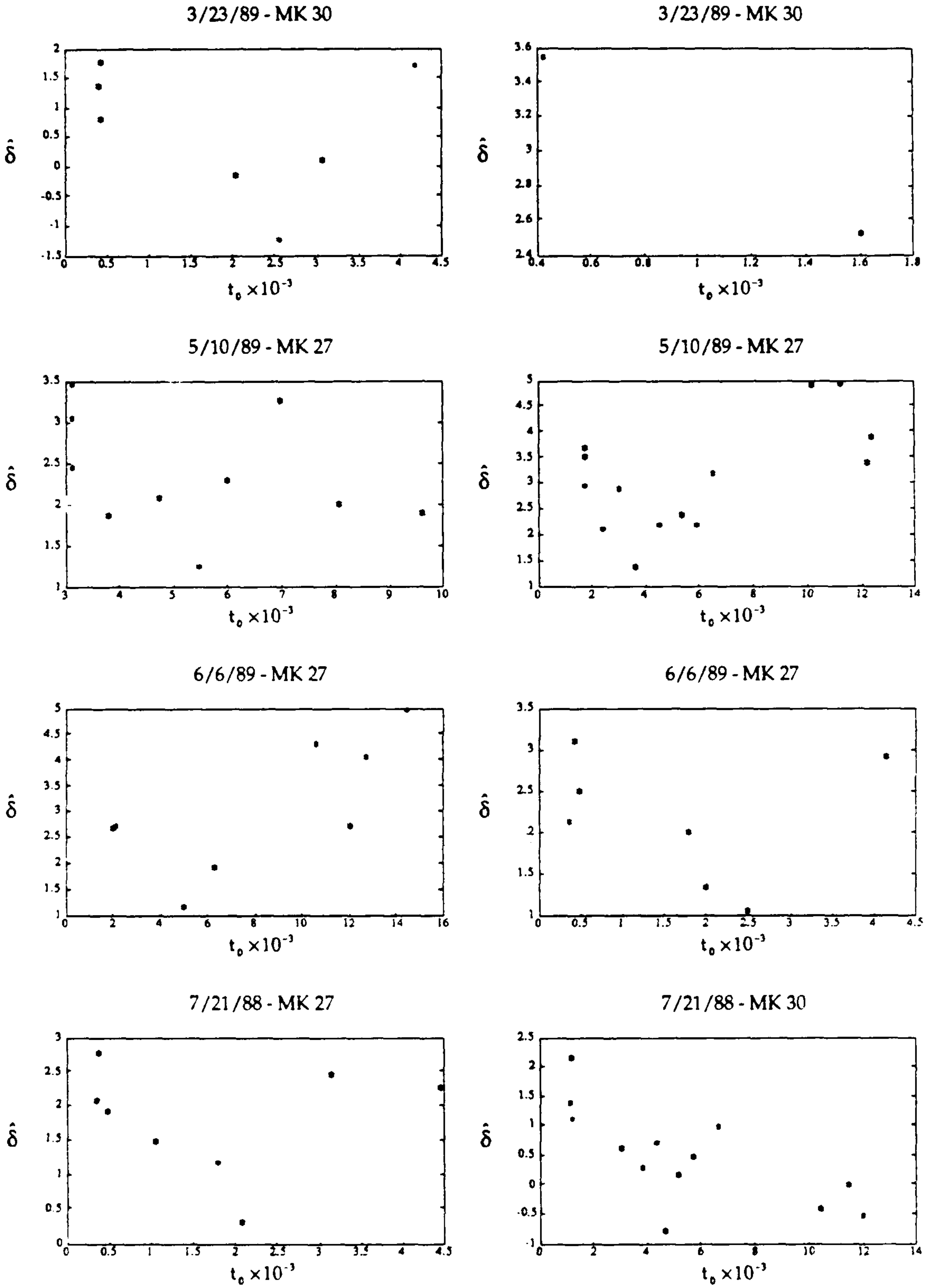


Figure 4. Offset vs. Central Point Count for the eight runs

Since the $\hat{\delta}$ values are significant, it is important to discover what they really represent. They have been labeled **timing synchronization offset errors** and, if that labeling is a valid physical description, then each run (vehicle) on each day should have its own unique value and this common value could be used to correct all track, not just crossover track. What follows is an analysis of this point.

Table 2 contains a graphical positioning of the sixty-nine $\hat{\delta}$ values according to the eight runs. There were eleven array pairs involved in this process and they are marked with the letters a, b, ..., k. (The correspondence with the overlap regions marked in Figure 2 is given in Table 3.) Study of Table 2 shows considerable "within run" variability, and one is inclined to doubt the reality of the constant offset interpretation.

TABLE 3. IDENTIFICATION OF ARRAY PAIRS TO RECONCILE TABLE 2 WITH FIGURE 2

a (1,11)	e (3,4)	i (5,56)
b (1,2)	f (4,5)	j (5,14)
c (2,11)	g (5,6)	k (15,16)
d (2,3)	h (4,55)	

In passing we also note that the $\hat{\delta}$ values for each array pair appear to have some temporal coherence, and hence the assignable causes may be related to this, but that is an issue for another time.

We proceed to model the components of random error variance and develop a test statistic for examining whether or not all $\hat{\delta}$ for the same run can be viewed as constant. The $\{\hat{\delta}_i\}$ are modeled as being affected by the day (water depth velocity profiles change from day to day), the run (the second run is a different vehicle operating later in the day), and the array pairs generating the crossover data. The array pairs are assumed to produce fixed effects, with values α_j for j=1, ..., 11; and the day and run effects are assumed to be random with zero means and variances σ_1^2 and σ_2^2 respectively. There is also an experimental (residual) error having zero mean and variance σ^2.

The result is the mixed model

$$\hat{\delta} = X\alpha + Z\beta + e \tag{20}$$

where $\hat{\delta}$ is the 69 component column vector of δ estimates; X is a 69 x 11 matrix of zeros and ones (incidence matrix) relating the $\hat{\delta}$ component to the array pair; α is the 11 component vector of array pair fixed effects; Z is a 69 rowed partitioned incidence matrix

$$Z = (Z_1, Z_2) \tag{21}$$

with Z_1 having 4 columns relating the $\hat{\delta}$ component to its day and Z_2 having 8 columns relating the $\hat{\delta}$ component to its run. The vector $\beta' = (\beta_1', \beta_2')$ represents the random variables

$$\beta_1' = (\beta_{11}, \ldots, \beta_{14}) \quad \text{independent} \quad N(0,\sigma_1^2)$$

$$\beta_2' = (\beta_{21}, \ldots, \beta_{28}) \quad \text{independent} \quad N(0,\sigma_2^2)$$

and the residuals are

$$e' = (e_1, \ldots, e_{69}) \quad \text{independent} \quad N(0,\sigma^2)$$

Further the vectors β_1, β_2, e are assumed to be mutually independent.

We will be applying the model (20) separately to each of the eight runs for purposes of estimating the fixed effects (α) for each of those runs. Such estimates will require values for the variance components ($\sigma^2, \sigma_1^2, \sigma_2^2$). We prefer to estimate them but once, by pooling all of the data. This can be accomplished separately from the estimation of fixed effects by using the restricted maximum likelihood method: the unequal block size version has been treated by Patterson and Thompson, Ref. [4] and is used here. (See Appendix A for a description of the algorithm.) The results are

$$\hat{\sigma}_1^2 = 0.3571 \qquad \hat{\sigma}_2^2 = 0.7207 \qquad \hat{\sigma}^2 = 0.2430 \tag{22}$$

and the units are seconds squared. It is noted that the run-to-run variance is double that of the day-to-day variance, which in turn is larger than the error variance by a ratio of about seven to five. But the degrees of freedom for the first two variances are small and the estimates may not be very reliable.

Using the model (20) one sees that the covariance matrix of the observables is

$$H = \sigma^2 I_n + Z\Gamma Z' \tag{23}$$

where I_n is the identity matrix of order n, Γ is diagonal, the first four values being σ_1^2 and the next eight σ_2^2; and Z is partitioned (Z_1, Z_2) as before. For the full data set n = 69, but recall that our goal is to check whether the fixed effects (offset estimates) can be viewed as constant within each run. The plan is to estimate the fixed effects for each run and test whether they can be viewed as constant (over the array pairs).

To do this we proceed as follows. The $\hat{\delta}$ values for the k^{th} run can be identified by the ones in the k^{th} column of Z_2, k = 1, ..., 8. The array pairs involved (not necessarily all eleven) are identified by ones in the X matrix restricted to the k^{th} run. The Z matrix is restricted in a like fashion and the covariance matrix (of order n, the number of crossover data sets in the k^{th} run) has the same structure as (23) with the reinterpretation of inputs. Of course the estimates (22) must also be input.

Let K be the number of array pairs involved in the k^{th} run. Use the Aitken estimator [3]

$$\hat{\alpha}_k = \left(X_k' \hat{H}_k^{-1} X_k\right)^{-1} X_k' \hat{H}_k^{-1} \hat{\delta}_k \tag{24}$$

and its covariance matrix

$$\mathrm{var}(\hat{\alpha}_k) = \left[X_k' \hat{H}_k^{-1} X_k\right]^{-1} = \hat{V}^{-1} \tag{25}$$

and the subscript k modifies the previous definitions of symbols so that only the k^{th} run is involved. If the null hypothesis is true (i.e., the α_k all fall on the main diagonal of the K dimensional space), then the maximum likelihood estimate of the common value is

$$\overline{\alpha} = \sum_{i=1}^{K} \sum_{j=1}^{K} V^{ij} \hat{\alpha}_k(j) / \sum_{i}^{K} \sum_{j}^{K} V^{ij} \tag{26}$$

It also follows from the normality assumptions that the quadratic

$$\left(\hat{\alpha}_k - \overline{\alpha}\right)' V\left(\hat{\alpha}_k - \overline{\alpha}\right) \quad \sim \quad \text{Chi Square } (K-1) \tag{27}$$

and this statistic is the weighted distance of the components of $\hat{\alpha}$ from the main diagonal (i.e., the constant fixed effect that is the same for all array pairs). Thus the null hypothesis should be rejected when this distance is too large.

The numerical results for the eight runs are contained in Table 4. The column marked p* contains the p values (empirical significance level) corresponding to the test statistics listed under the "distance" column. Seven of the eight values indicate rather rare events and the eighth, p* = 0.14, is associated with one degree of freedom. Low degrees of freedom tend to dampen the chances of finding significant results. The final column is the ratio of the distance (27) to its standard deviation under the null hypothesis.

We assert that the array pair effects are not constant and that there are other sources of systematic error that dominate this process.

TABLE 4. TESTING THE EIGHT RUNS FOR CONSTANT FIXED EFFECTS

$\overline{\alpha}$	K−1	Distance	p*	distance/SD
0.61	6	30.17	3.6×10^{-5}	8.71
3.03	1	2.12	0.14	1.51
2.35	8	17.71	0.06	4.43
3.09	9	51.67	5.2×10^{-8}	12.18
3.05	4	41.48	2.1×10^{-8}	14.67
2.13	6	14.16	0.03	4.09
1.79	7	18.13	0.01	4.85
0.46	8	30.44	1.8×10^{-4}	7.61

4. CHARACTERISTICS OF NOISE

In order to study the stochastic nature of the tracking errors we fit straight lines to the track. (This was done even for track that did not appear linear via a visual scan of graphical output. The exceptional cases are marked in Appendix C.) It is assumed that the target vehicles speed is constant. The result is a set of deterministically spaced points that fall on a straight line in three space. From these we can produce residuals and, assuming local stationarity, study the noise structure.

The basic (unadjusted for constant speed) straight line is found by principal components. Let X(t) be used for X(t,0), eq. (6) and

$$\overline{X} = \frac{1}{T}\sum_{t} X(t) \tag{28}$$

We seek a projection (direction) $p = (p_1, p_2, p_3)'$

$$Z(t) = \sum_{1}^{3} p_i X_i(t) = p'X(t) \tag{29}$$

that will maximize the variance of {Z(t)}

$$\sigma_Z^2 = \frac{1}{T}\sum_{t}\left[Z(t) - \overline{Z}\right]^2 = p'C_x p \tag{30}$$

where

$$C_x = \frac{1}{T}\sum_{t}\left(X(t) - \overline{X}\right)\left(X(t) - \overline{X}\right)' \tag{31}$$

is the covariance matrix of the track data, X(t).

Since the vector p is a set of direction numbers, we adopt the usual constraint, $\sum_1^3 p_j^2 = 1$. (This will keep σ_Z^2 finite.) It is easily shown that the three solutions to the eigen problem

$$C_x p = \lambda p$$

provide us with an orthonormal basis

$$P = \{P_1, P_2, P_3\} \tag{32}$$

where P_i is the eigen vector corresponding to λ_i and $\lambda_1 > \lambda_2 > \lambda_3$. Since $\sigma_Z^2 = p'C_x p = \lambda p'p = \lambda$, and our goal is to maximize σ_Z^2, we choose the first eigen vector for use.

The set of values

$$Z_1(t) = P_1'\left(X(t) - \overline{X}\right) \tag{33}$$

represents the succession of values after the vectors $\left\{X(t) - \overline{X}\right\}$ are projected onto the line of the first principal component. These projections are orthogonal projections and may not well represent the relative positions of a target vehicle heading in the direction P_1 at constant speed. Accordingly it is appropriate to make an adjustment in the $\{Z_1(t)\}$ to account for constant speed. The collection of point counts,

$$t_1 < t_2 < \ldots < t_T, \tag{34}$$

all differ by integral multiples of some base value, Δ, representing distance traversed per point count. Let us perform a simple linear regression of the $Z_1(t)$ on the times (34). The fitted values are

$$\hat{Z}(t) = a + b\left(t - \bar{t}\right) \tag{35}$$

where $a = \frac{1}{T}\sum_t Z_1(t)$ and $b = \sum_t \left(t - \bar{t}\right)Z_1(t) / \sum_t \left(t - \bar{t}\right)^2$. Now the successive values of $\hat{Z}(t)$ differ by integral multiples of a converted base value, $b\Delta$, and represent the constant speed progression of the vehicle in the direction P_1.

It remains to represent the distances $\hat{Z}(t)$ in the original coordinate system. Let

$$R(t) = \left(\hat{Z}(t), 0, 0\right)' \tag{36}$$

be the estimated position of the vehicle in the basis P, and then

$$X_{sl}(t) = \overline{X} + PR(t) \tag{37}$$

will be the straight line in the original coordinates. Similarly, when $\{Y(t)\}$ is put into this algorithm, we produce $Y_{sl}(t)$.

Now we are positioned to estimate the covariance matrices of the noise processes

$$D_x = \frac{1}{T}\sum_t [X(t) - X_{sl}(t)][X(t) - X_{sl}(t)]'$$

$$D_y = \frac{1}{T}\sum_t [Y(t) - Y_{sl}(t)][Y(t) - Y_{sl}(t)]' \tag{38}$$

$$D_{xy} = \frac{1}{T}\sum_t [X(t) - X_{sl}(t)][Y(t) - T_{sl}(t)]'$$

Computational work shows considerable variability in these matrices as one changes the day, the run, and the array pair. Also the cross correlations are mostly different from zero. The matrix M, eq. (16), is estimated by

$$\hat{M} = D_x + D_y - D_{xy} - D_{xy}' \tag{39}$$

and used in (17), (18), and (19). The quantities (38) and (39) appear in Appendix C, and illustrate their variable nature.

Another immediate use of the noise processes is to look at the autocorrelations. In addition to the three individual components we define a "noise displacement" process

$$d(t) = \left\{\sum_1^3 \left(X_i(t) - X_{sl_i}(t)\right)^2\right\}^{\frac{1}{2}} \tag{40}$$

(Of course the same is done for $\{Y(t)\}$.)

The autocovariances

$$R(h) = \frac{1}{T} \sum_{t=1}^{T-h} \left(d(t) - \bar{d}\right)\left(d(t+h) - \bar{d}\right) \tag{41}$$

for $\{d(t)\}$ and for the individual components $\{X_i(t) - X_{sl_i}(t)\}$ are computed and normalized. Plots of R(h)/R(0) vs h appear in Appendix D. It is typical that they become and remain in a general level of "static" for $h \geq 1$.

5. SUMMARY

The paper contains methodology for estimating the presence of timing synchronization error and judging its significance in the framework of our short baseline array underwater test range. The methodology was applied to real data. Some plots illustrating the effects appear in Appendix B. One must view them with care because the coordinate scales are so different. It is rather typical that some unresolved systematic error is exposed in the side view.

Analysis of the results does not support the timing synchronization error model as accounting for the discrepancies. Other sources of systematic error must be unmasked first. Possibilities include the orientation of the arrays [2], biases in the raytracing inputs [3], and perhaps a temporal or spatial gradient in the depth velocity profile.

Appendices C and D contain information about the second order properties of the three dimensional noise process and about time lag correlations.

APPENDIX A. PATTERSON AND THOMPSON ALGORITHM

The purpose of this appendix is to document the computational method for estimating variance components that is but implicitly described in the Patterson and Thompson paper [4]. We use the notation of that paper. Specifically

$$y = X\alpha + \epsilon \tag{A.1}$$

where the observeable y is an n component column vector, X is an n by t matrix of rank t determined by the allocation of treatments to units, α is a t vector of fixed effects, and ϵ is a mean zero n vector of normal random variables with covariance matrix

$$V = \sigma^2 H \qquad H = Z\Gamma Z' + I. \tag{A.2}$$

The matrix I is the identity of order n, Z is the n by b design matrix for c block factors, and Γ is a diagonal matrix containing the variance components relative to the basic error variance σ^2. I.e.,

$$H = I + \sum_{p=1}^{c} Z_p Z_p' \gamma_p \tag{A.3}$$

and each block design matrix, Z_p, is n by b_p; each plot having exactly one level in each Z_p, p = 1, ..., c, the variance of the p^{th} block is

$$\sigma_p^2 = \gamma_p \sigma^2 \qquad \text{for } p = 1, 2, \ldots, c \tag{A.4}$$

and Z in (A.2) has the partitioned form

$$Z = (Z_1, Z_2, \ldots, Z_c). \tag{A.5}$$

The diagonal elements of Γ are b_1 consecutive γ_1's, b_2 consecutive γ_2's, ..., and b_c consecutive γ_c's, with $\sum_{p=1}^{c} b_p = b$.

Out goal is the estimation of $\gamma_1, \gamma_2, ..., \gamma_c, \sigma^2$ by the restricted maximum likelihood method for the unbalanced case. The steps follow. Let

$$S = I - X(X'X)^{-1}X' \tag{A.6}$$

$$W = Z'SZ + \Gamma^{-1} \tag{A.7}$$

The algorithm is an iterative one, and initial values for the $\gamma_1, ..., \gamma_c$ are needed to develop W in (A.7). The b by b matrix W and its inverse can be partitioned according to $b_1, b_2, ..., b_c$. So doing allows both

$$\beta = W^{-1}Z'Sy \tag{A.8}$$

$$U = \Gamma^{-1} - \Gamma^{-1}W^{-1}\Gamma^{-1} \tag{A.9}$$

to be viewed as partitioned. I.e.,

$$\begin{aligned} \beta' &= \left(\beta_1', \beta_2', \ldots, \beta_c'\right) \\ U &= \{U_{ij}\} \quad i, j = 1, \ldots, c \end{aligned} \tag{A.10}$$

and U_{ij} is a b_i by b_j matrix.

Now we are positioned to define

$$\begin{aligned} R &= y'Sy - y'SZ\beta \\ \beta_p &= \beta_p' \beta_p / \gamma_p^2 \end{aligned} \tag{A.11}$$

and E_p = trace (U_{pp}) for p = 1, ..., c. Then define the modified information matrix $\{f_{ij}\}$, of order c+1,

$$\begin{aligned} f_{ij} &= \text{trace}\,(U_{ij}U_{ji}) \quad \text{for } i, j = 1, \ldots, c \\ f_{p,c+1} &= f_{c+1,p} = E_p \quad \text{for } p = 1, \ldots, c \\ f_{c+1,c+1} &= n-t \end{aligned} \tag{A.12}$$

and let $\{f^{ij}\}$ be its inverse.

Finally, the estimator update equations are, using $k = c+1$

$$\begin{aligned} \hat{\sigma}^2 &= \sum_{j=1}^{k-1} f^{kj}B_j + f^{kk}R \\ \hat{\gamma}_i &= \hat{\gamma}_i^{(0)} + \frac{1}{\hat{\sigma}^2}\left\{\sum_{j=1}^{k-1} f^{ij}B_j + f^{ik}R\right\} \quad \text{for } i = 1,\ldots,c \end{aligned} \tag{A.13}$$

and the new $\{\hat{\gamma}_i\}$ can be placed in (A.7) to start the next iteration. No initial value for σ^2 is required, but it must be updated at each iteration. The algorithm should be stopped when (A.13) is stable. The variance components are computed from (A.4)

The application in the present report has c=2, with $b_1 = 4$ and $b_2 = 8$. It is helpful to record the partitioning and inversion of W, (A.7).

$$\begin{aligned} W &= \begin{bmatrix} Z_1'SZ_1 & Z_1'SZ_2 \\ Z_2'SZ_1 & Z_2'SZ_2 \end{bmatrix} + \begin{bmatrix} I_1/\gamma_1 & 0 \\ 0 & I_2/\gamma_2 \end{bmatrix} \\ &= \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \qquad \text{symmetric.} \end{aligned} \tag{A.14}$$

$$W^{-1} = \begin{bmatrix} W^{11} & W^{12} \\ W^{21} & W^{22} \end{bmatrix} \tag{A.15}$$

can be computed from

$$W^{22} = \left[W_{22} - W_{21}W_{11}^{-1}W_{12}\right]^{-1}$$

$$W^{12} = -W_{11}^{-1}W_{12}W^{22} \tag{A.16}$$

$$W^{11} = \left[I_1 - W^{12}W_{21}\right]W_{11}^{-1}$$

APPENDIX B. EFFECTS OF CORRECTIONS

Some selected results of applying the method are given in Figures B1 through B4; two solutions for each of the four days, top and side views for each. Original track from the first member of the array pair is marked with a cross, and from the second member with a small circle. The (timing) corrected tracks are marked with connected solid lines for the former and dashed lines for the latter.

The first set in Figure B1 shows no noticeable corrections, $\hat{\delta}$ is small. The second set in that figure has $\hat{\delta} = 3.54$ and is a real data version of the situation depicted by Figure 3. The first set of Figure B2 appears as if one track is corrected more than the other. But this can be explained because the "stretching angles" are different. The second set of this figure shows a rather common condition in that the corrected track is at a shallower depth than the original.

The first set in Figure B3 has something of a "showcase" flavor; the top view has corrected track with desirable coherence. (But a systematic separation in the vertical remains.) The second set also shows tantalizing improvement. The first set in Figure B4 has $\hat{\delta} = -0.8$, and provides an example of modest "contracting" of points. The second set is an extreme case of curved track. The top view has an excellent correction displayed, but the side shows that the vertical is still separated.

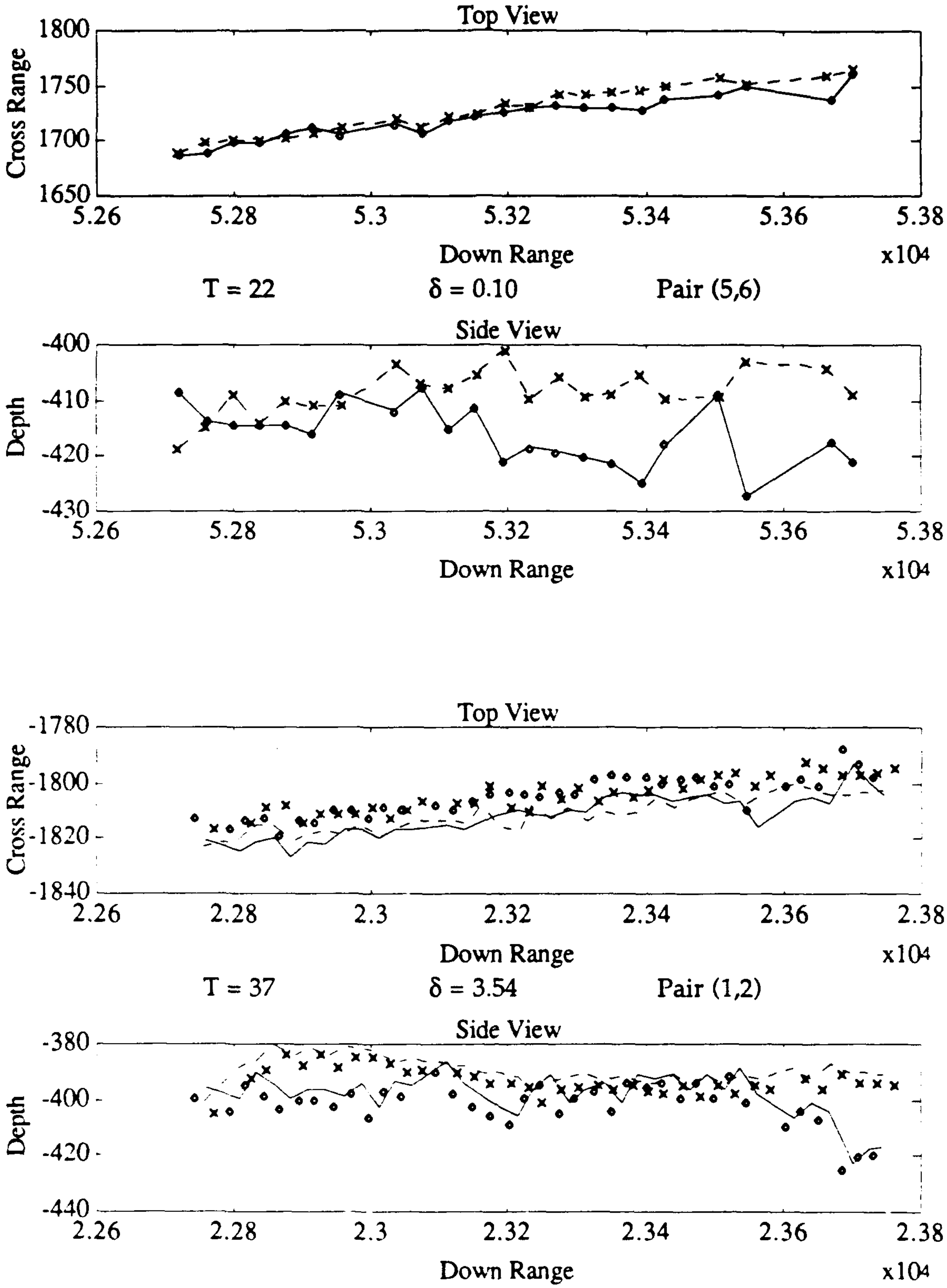


Figure B1. Before and After Offset Correction - 3/23/89

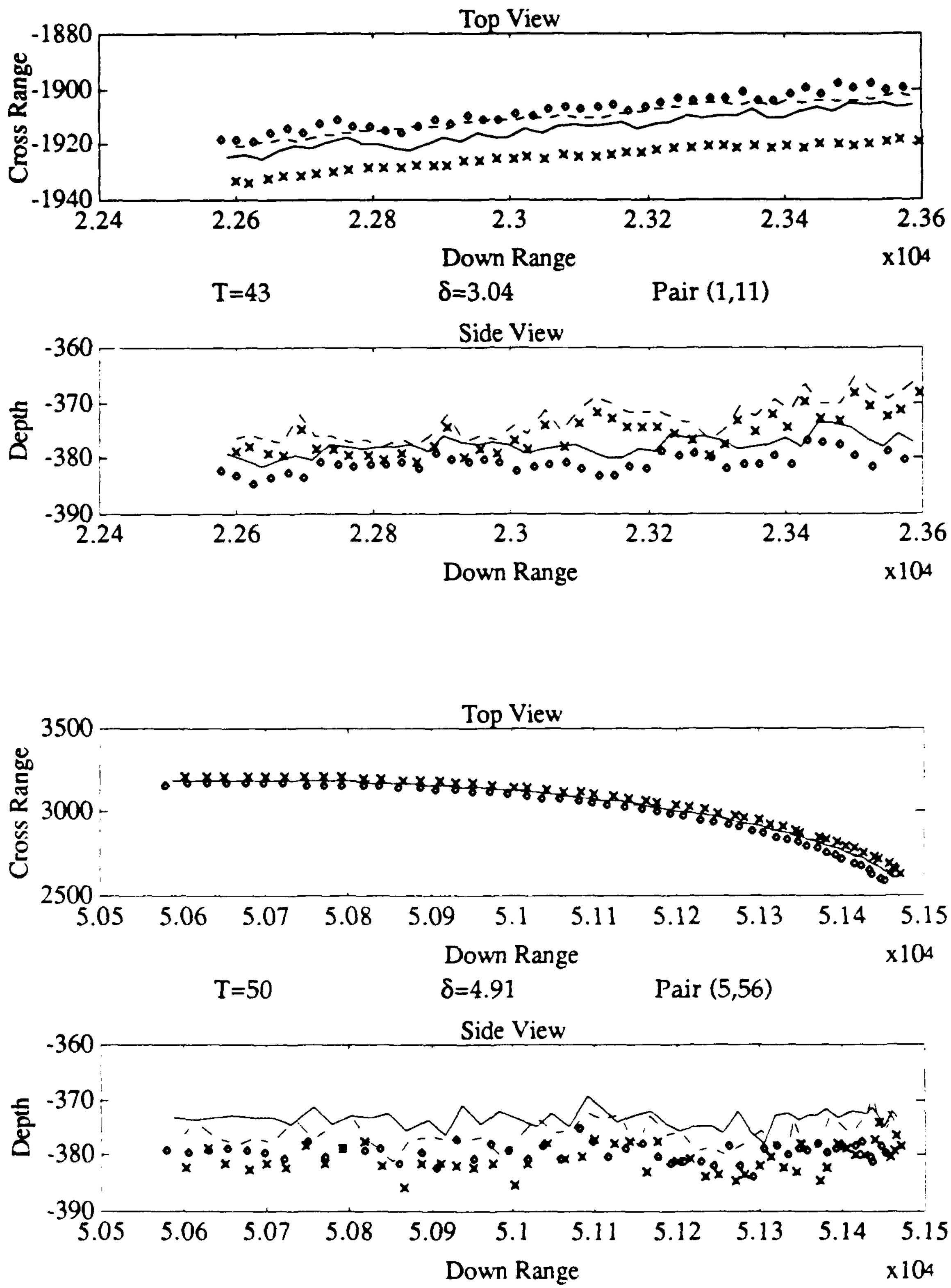


Figure B2. Before and After Offset Correction - 5/10/89

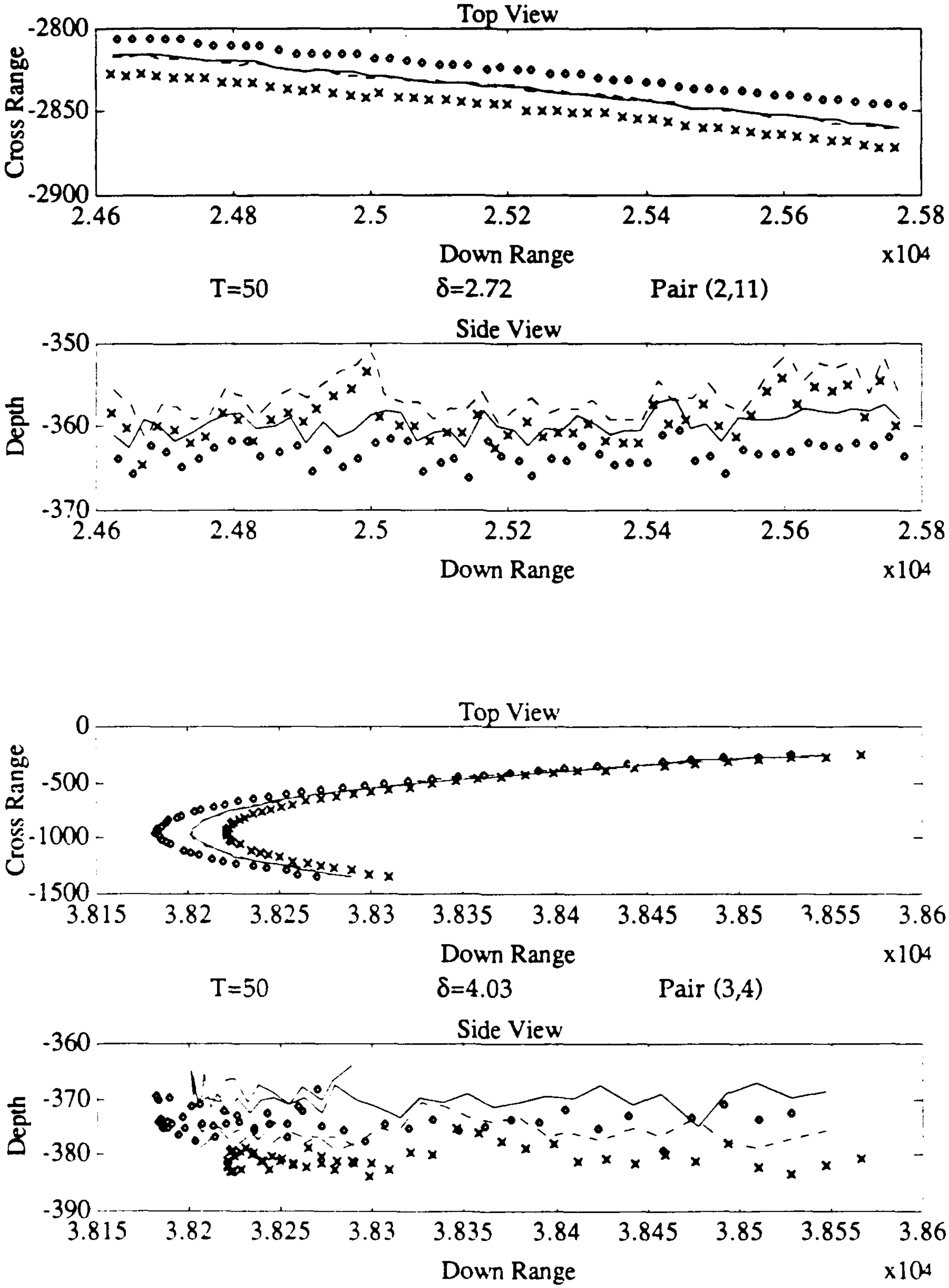
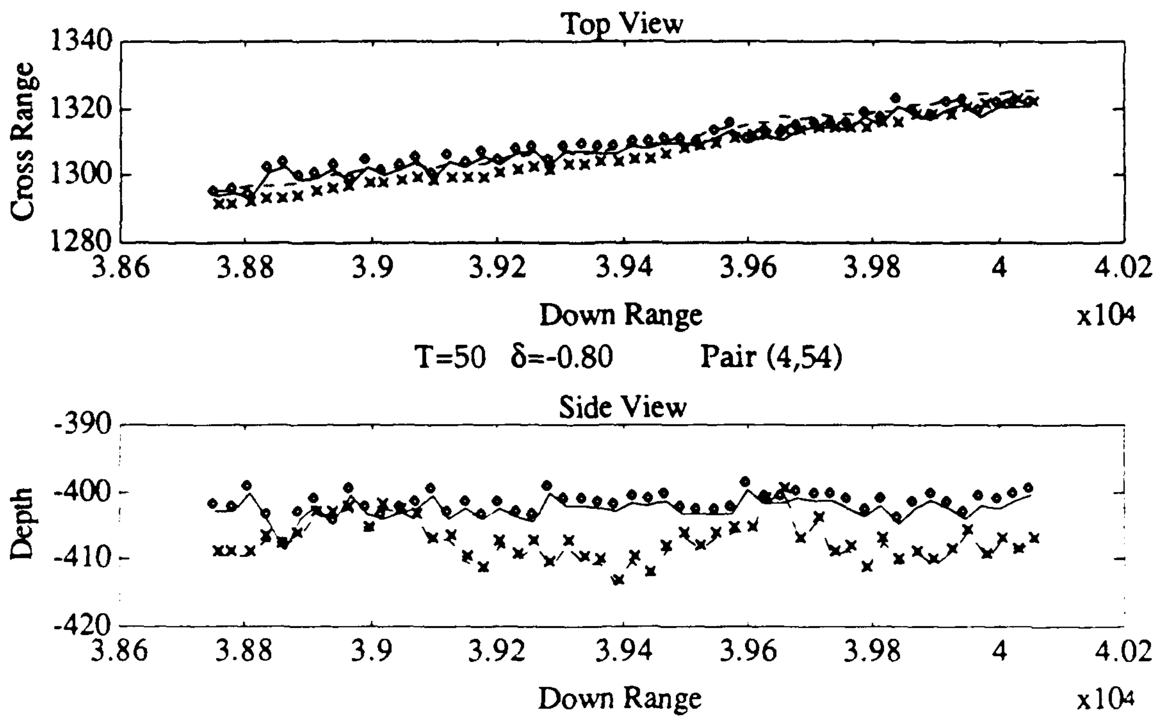


Figure B3. Before and After Offset Correction - 6/6/89



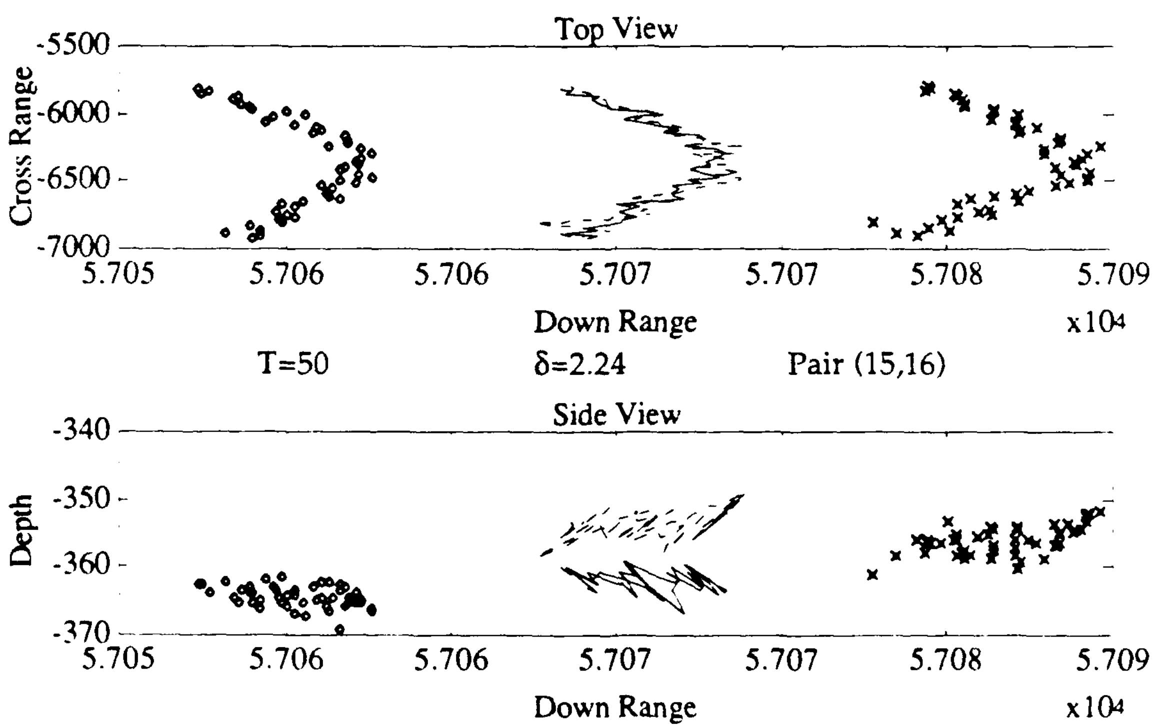


Figure B4. Before and After Offset Correction - 7/21/89

APPENDIX C. COVARIANCE MATRICES

The covariances of the residual processes, eq. (38), are recorded here along with $\hat{M}$, eq (39), which estimates M, eq. (16). The cross covariances D_{xy} are converted to correlations, R_{xy}, by

$$R_{xy}(i,j) \;=\; D_{xy}(i,j) / \left[D_x(i,i) D_y(j,j)\right]^{\frac{1}{2}}; \quad i,j = 1,2,3.$$

The order of presentation is that of Table 1. The number of points used, T, is occasionally smaller than its Table 1 counterpart. This is due to outlier rejection based upon a visual view of the track. In three instances these values are zero and then, the covariances are not computed.

In a number of instances the covariances are marked with a "c" or "cc" in the identifying column. This means the track was curved (c) or excessively curved (cc) so that the straight line filter could not reasonably be assumed to provide a valid representation of the path. It is a curiosity that, in many of these instances, the cross correlations are strong and their use produces, via eq. (16), compensatory values. That is, the M matrices do not appear to be especially large. Some large cross correlations also appear in cases that pass the visual test for straight line track.

Table C.1. Covariances and Cross Correlations of Residuals

Date	T	Dx			Dy			Rxy			M		
3/23/89 MK-30	42	12.91	1.46	-5.27	5.47	4.01	-4.69	.39	.85	-.12	11.84	2.65	-7.93
		1.46	1.24	-5.06	4.01	6.43	-.46	.07	.12	-.08	2.65	6.97	-3.88
		-5.27	-5.06	23.17	-4.69	-.46	22.77	-.04	-.13	.21	-7.93	-3.88	36.51
	33	4.26	-4.82	6.59	4.06	4.42	-7.29	.47	.31	-.49	4.43	-.50	-.42
		-4.82	7.50	-4.08	4.42	7.06	-3.50	-.29	-.25	.23	-.50	18.26	-12.38
		6.59	-4.08	16.36	-7.29	-3.50	23.17	.37	.18	-.48	-.42	-12.38	58.27
	19	14.29	-24.55	13.39	2.23	4.50	-1.62	.26	.48	.12	13.55	-22.53	9.19
		-24.55	49.68	-11.93	4.60	14.02	4.31	-.17	-.33	-.16	22.53	81.28	-8.68
		13.39	-11.93	29.00	-1.62	4.31	14.98	.11	.16	-.10	9.19	-8.68	47.95
	45	11.73	.49	-3.60	4.17	-5.00	4.84	.32	-.82	.53	11.46	-3.24	-2.48
		.49	.71	-2.89	-5.00	7.78	-2.83	.02	.01	.09	-3.24	8.47	-6.21
		-3.60	-2.89	13.00	4.84	-2.83	12.36	-.01	-.14	-.20	-2.48	-6.21	30.39
	41	2.37	3.36	-1.75	16.93	1.77	2.34	.06	.01	.07	18.56	6.17	.09
		3.36	11.96	10.21	1.77	1.37	-4.09	-.61	-.10	.52	6.17	14.13	3.65
		-1.75	10.21	26.59	2.34	-4.09	19.65	-.15	-.04	.11	.09	3.65	41.39
	22	4.10	6.89	3.70	5.40	-10.92	-1.90	-.23	.43	-.26	11.63	-5.77	4.49
		6.89	14.24	1.59	-10.92	28.52	-7.00	-.18	.33	-.18	-5.77	29.41	-5.05
		3.70	1.59	[illegible]	-1.90	-7.00	19.49	-.07	.17	-.18	4.49	-5.05	36.67
	21	5.17	-11.92	8.60	5.38	2.38	-17.76	.08	-.00	-.07	9.70	-8.10	-7.91
C		-11.92	30.49	-18.51	2.38	14.14	-1.35	-.16	-.02	.33	-8.10	45.39	-25.45
		8.60	-18.51	16.16	-17.76	-1.35	68.60	-.03	.01	.09	-7.91	-25.45	78.93
3/23/89 MK-30	37	2.56	-2.78	4.49	5.52	5.26	-13.34	.25	-.12	-.48	6.22	4.57	-4.08
		-2.78	6.39	.62	5.26	12.43	-1.50	-.25	-.11	.27	4.57	20.70	-1.07
		4.49	.62	18.37	-13.34	-1.50	49.73	.00	-.22	-.22	-4.08	-1.07	81.51
	47	.92	-1.52	1.18	10.33	17.40	-14.20	-.14	-.10	.11	12.11	15.42	-13.82
		-1.52	4.53	.98	17.40	32.04	-20.33	.21	.19	-.11	15.42	[illegible]	-18.64
		1.18	.98	6.06	-14.20	-20.33	25.14	-.02	.08	.09	13.82	-18.64	29.04
5/10/89 MK-27	43	3.77	.50	-.38	2.30	.96	-.13	.31	-.35	.02	4.25	1.00	-.11
		.50	.89	-1.57	.96	1.82	.16	.36	.03	.09	1.00	2.64	-2.13
		-.38	-1.57	4.86	-.13	.16	1.82	-.18	.34	-.29	-.11	-2.13	8.39
	38	1.14	-1.25	1.38	1.09	.87	-1.08	.53	.29	-.36	1.04	-1.07	.30
		-1.25	3.21	-.86	.87	1.46	-.17	.12	-.08	.11	-1.07	5.03	-2.13
		1.38	-.86	3.11	-1.08	-.17	2.40	.40	.31	-.22	.30	-2.13	6.74
	25	2.05	-3.96	.96	.48	.68	-.45	-.26	-.31	.14	3.06	-2.90	1.04
		-3.96	9.36	1.44	.68	3.02	1.87	.11	.26	.05	-2.90	9.66	3.40
		.96	1.44	7.37	-.45	1.87	4.08	-.27	-.05	.32	1.04	3.40	7.98
	34	2.93	-.16	.32	1.11	-1.48	1.31	-.07	.36	.11	4.31	-1.77	.59
		-.16	.45	-1.15	-1.48	3.33	-1.09	-.04	.43	-.08	-1.77	2.73	-.96
		.32	-1.15	4.75	1.31	-1.09	3.13	.26	-.79	.38	.59	-.96	4.99

Date	T	Dx			Dy			Rxy			M		
	56	2.63	-3.61	1.94	1.93	1.84	-1.72	.32	-.33	.05	3.12	-.06	.09
		-3.61	6.23	-1.96	1.84	5.70	-2.40	-.15	.18	-.11	-.06	9.75	-2.85
		1.94	-1.96	4.08	-1.72	-2.40	3.09	-.01	-.22	.01	.09	-2.85	7.07
	59	2.29	-2.16	3.26	1.35	.28	-.81	.33	-.40	.31	2.49	-.95	1.48
		-2.16	4.59	-1.95	.28	2.78	-1.14	.03	.21	-.01	-.95	5.86	-1.81
		3.26	-1.95	7.30	-.81	-1.14	4.27	-.01	-.28	.40	1.48	-1.81	7.06
	50	132.12	525.44	-7.44	109.42	465.46	9.12	.96	.79	.59	10.21	5.64	-4.12
CC		525.44	3906.70	-7.16	465.46	827.65	47.43	.66	1.00	.40	5.64	5.19	-1.34
		-7.44	-7.16	4.25	9.12	47.43	4.47	-.29	-.07	-.15	-4.12	-1.34	9.99
	50	2.04	-.72	-.62	6.07	-2.69	2.32	.31	-.16	.05	5.92	-2.69	1.66
		-.72	.70	-.91	-2.69	1.52	-.44	-.22	.23	-.01	-2.69	1.75	-1.36
		-.62	-.91	4.18	2.32	-.44	3.03	-.08	.02	-.08	1.66	-1.36	7.74
	40	.68	1.32	1.78	.58	-1.05	-1.58	.11	.27	-.06	1.12	-.75	.42
		1.32	4.55	2.41	-1.05	12.35	-1.59	.20	-.07	-.13	-.75	17.94	.67
		1.78	2.41	5.36	-1.58	-1.59	6.87	-.05	.22	.11	.42	.67	10.88
5/10/89	50	23.19	22.18	16.56	9.27	-.90	.62	.38	.07	-.02	21.40	20.37	20.41
MK-27		22.18	33.81	15.26	-.90	7.03	-1.82	-.03	.27	-.01	20.37	32.48	14.03
		16.56	15.26	27.68	.62	-1.82	3.29	-.32	-.04	-.00	20.41	14.03	31.01
	45	2.93	.38	-.86	.83	.73	-1.86	.19	.07	-.51	3.17	1.00	-1.55
		.38	.63	-1.82	.73	1.35	-.69	.03	.27	.00	1.00	1.48	-2.47
		-.86	-1.82	7.55	-1.86	-.69	5.81	.03	-.03	-.11	-1.55	-2.47	14.79
	37	1.00	-1.27	1.11	.99	1.06	-2.03	.00	.11	.10	1.99	-.69	-1.19
		-1.27	2.91	-.18	1.06	1.77	-1.38	.22	.13	-.35	-.69	4.10	-1.36
		1.11	-.18	2.81	-2.03	-1.38	5.54	.04	.14	.06	-1.19	-1.36	7.84
	45	.74	1.71	.93	.69	.05	-2.72	.08	-.08	.01	1.31	2.16	-2.12
		1.71	8.30	-2.18	.05	2.17	-2.77	-.17	-.06	.40	2.16	10.97	-5.93
		.93	-2.18	6.07	-2.72	-2.77	14.07	.10	-.04	-.09	-2.12	-5.93	21.89
	33	1.94	.24	-.80	.91	-1.33	.92	-.09	.66	.09	3.10	-1.37	-1.18
		.24	.50	-1.60	-1.33	2.91	-.31	-.07	.40	-.01	-1.37	2.45	-.30
		-.80	-1.60	7.16	.92	-.31	2.46	.49	-1.39	.25	-1.18	-.30	7.49
	37	2.17	-3.28	2.44	1.56	2.19	-2.19	-.21	-.19	.24	4.52	-1.28	.11
		-3.28	5.61	-2.96	2.19	3.79	-2.00	.26	.23	-.32	-1.28	7.32	-3.01
		2.44	-2.96	4.85	-2.19	-2.00	4.79	-.16	-.11	.19	.11	-3.01	7.81
	50	.41	-.69	.52	1.49	2.06	-2.61	-.25	-.07	.19	2.29	1.38	-2.10
		-.69	2.87	1.15	2.06	3.90	-2.37	.12	.10	-.03	1.38	6.12	-1.31
		.52	1.15	3.68	-2.61	-2.37	6.33	-.27	.05	.34	-2.10	-1.31	6.74
	0	---	---	--	---	---	---	---	---	---	---	---	---
C		---	---	---	---	---	---	---	---	---	---	---	---
		---	---	---	---	---	---	---	---	---	---	---	---
	0	---	---	---	---	---	---	---	---	---	---	---	---
C		---	---	---	---	---	---	---	---	---	---	---	---
		---	---	---	---	---	---	---	---	---	---	---	---

Date	T	Dx			Dy			Rxy			M		
	39	2.04	1.11	.17	1.83	2.21	-.60	.15	.01	.14	3.28	2.75	-1.26
		1.11	1.14	1.39	2.21	3.08	-1.51	.22	.22	.11	2.75	3.39	-.81
		.17	1.39	3.87	-.60	-1.51	1.96	.23	.22	.20	-1.26	-.81	4.73
	55	3.17	-1.58	2.08	2.74	1.09	-1.15	.68	-.02	-.10	1.93	-.82	1.35
		-1.58	4.42	-2.51	1.09	2.82	.09	.13	.12	.37	-.82	6.37	-3.56
		2.08	-2.51	4.36	-1.15	.09	4.54	-.01	-.03	-.03	1.35	-3.56	9.16
	50	1.04	-1.43	3.91	.45	-.05	-1.64	.03	.03	-.05	1.45	-1.31	2.40
		-1.43	12.39	-4.40	-.05	1.96	-.06	-.16	.11	.10	-1.31	13.26	-5.50
		3.91	-4.40	15.02	-1.64	-.06	6.26	.00	.06	-.05	2.40	-5.50	22.23
	54	.59	.76	.92	.56	-.47	-.76	.19	.05	-.02	.94	.02	.54
		.76	4.47	-2.17	-.47	2.03	.15	.17	-.07	-.07	.02	6.92	-2.15
		.92	-2.17	6.72	-.76	.15	2.03	-.31	.12	.05	.54	-2.15	8.38
	21	8.84	25.18	-1.76	10.11	28.52	.14	.75	.84	.00	4.84	-3.15	-1.18
C		25.18	107.44	.08	28.52	111.27	-2.10	.93	.99	-.08	-3.15	2.49	1.54
		-1.76	.08	4.53	.14	-2.10	1.66	-.11	-.13	-.36	-1.18	1.54	8.15
	46	1.80	2.60	2.80	.76	-1.62	-.26	-.01	-.24	.35	2.58	1.37	1.99
		2.60	5.00	2.22	-1.62	5.79	-2.58	.02	-.20	.21	1.37	12.90	-1.62
		2.80	2.22	7.20	-.26	-2.58	4.46	-.09	-.01	.22	1.99	-1.62	9.19
6/6/89	19												
MK-27		22.47	2.34	44.62	20.54	2.78	43.31	.92	.62	.99	3.37	.91	-4.02
C		2.34	1.20	2.35	2.78	2.26	7.06	.16	.48	.25	.91	1.88	.81
		44.62	2.35	108.62	43.31	7.06	108.13	.92	.41	.95	-4.02	.81	11.69
	50	.59	-.42	1.83	.27	-.01	-.71	.34	-.10	-.14	.59	-.26	1.05
		-.42	4.12	-1.34	-.01	2.32	.19	-.06	.08	.13	-.26	5.94	-1.41
		1.83	-1.34	6.30	-.71	.19	2.42	.21	-.07	-.13	1.05	-1.41	9.75
	41	2.14	-2.12	2.08	.69	1.37	.37	.35	.47	-.07	1.97	-1.12	2.09
		-2.12	3.03	-1.37	1.37	3.39	1.97	-.12	-.09	.16	-1.12	6.99	-1.20
		2.08	-1.37	4.66	.37	1.97	2.81	.19	.39	.09	2.09	-1.20	6.84
	50	1.92	-1.31	-.15	1.50	-.79	.96	.12	-.36	-.15	3.00	-1.04	.29
C		-1.31	1.57	-1.49	-.79	.76	-.04	-.42	.60	-.07	-1.04	1.03	-.98
		-.15	-1.49	5.17	.96	-.04	1.80	.50	-.25	.31	.29	-.98	5.09
	33	814.12	-28.74	-10.84	822.25	-23.79	1.99	1.00	-.28	.09	.73	.96	-1.07
C		-28.74	9.11	1.42	-23.79	7.32	-.89	-.38	.73	.14	.96	4.53	-.74
		-10.84	1.42	1.63	1.99	-.89	4.39	-.19	.12	-.08	-1.07	-.74	6.45
	35	1.70	-2.22	2.57	1.83	3.08	-1.64	-.16	-.27	.01	4.12	1.26	1.25
		-2.22	3.77	-1.75	3.08	6.71	-1.07	.09	.16	.00	1.26	8.85	-2.25
		2.57	-1.75	7.55	-1.64	-1.07	5.50	-.12	-.13	-.07	1.25	-2.25	13.99
	35	.54	-.71	1.51	1.14	-.06	-4.60	-.08	.13	.18	1.81	-1.20	-3.35
		-.71	3.71	-1.40	-.06	5.61	1.32	.10	.08	-.15	-1.20	8.60	-.40
		1.51	-1.40	4.38	-4.60	1.32	18.88	-.05	.12	.12	-3.35	-.40	21.08
	51	.50	.02	1.25	1.04	-.80	-1.73	.39	-.09	-.03	.97	-.54	-.55
		.02	2.50	-1.64	-.80	2.09	.52	-.09	.01	-.03	-.54	4.53	-1.10
		1.25	-1.64	5.59	-1.73	.52	4.55	.09	.03	.14	-.55	-1.10	8.74

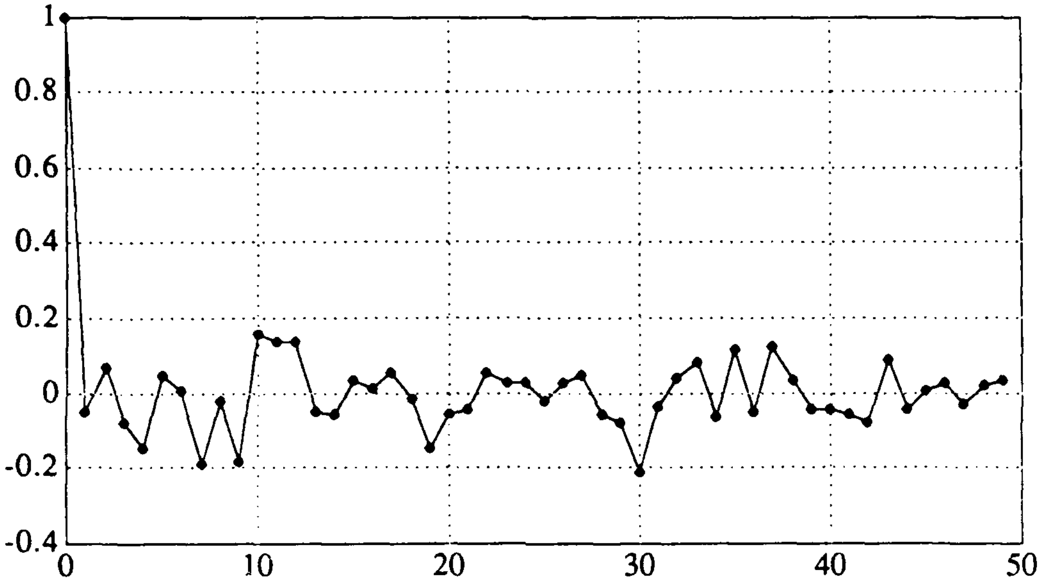
Date	T	Dx			Dy			Rxy			M		
6/6/89	50	5.64	1.24	-.54	1.87	1.04	-2.32	.18	-.29	.05	6.32	2.39	-2.57
MK-27		1.24	.56	-1.15	1.04	2.89	2.46	.05	-.02	-.04	2.39	3.50	1.85
		-.54	-1.15	5.18	-2.32	2.46	12.77	-.05	-.15	.08	-2.57	1.85	16.59
	24	2.11	-3.07	1.33	.96	.77	-2.12	-.01	-.16	-.60	3.11	-2.01	.03
		-3.07	4.83	-1.40	.77	1.70	.46	.03	.22	.80	-2.01	5.29	-2.59
		1.33	-1.40	2.33	-2.12	.46	9.15	.02	.01	-.09	.03	-2.59	12.28
	41	.94	-1.28	1.88	1.16	.07	-3.08	.24	.40	.10	1.60	-2.39	-1.00
C		-1.28	4.44	-3.49	.07	5.25	4.12	.13	-.12	-.09	-2.39	10.89	-.41
		1.88	-3.49	6.17	-3.08	4.12	14.24	-.13	.19	.17	-1.00	-.41	17.31
	50	5.12	-.37	-3.60	5.36	-4.96	3.52	-.18	.43	.07	12.33	-6.79	-.61
		-.37	.92	-1.39	-4.96	5.34	-3.31	.10	-.03	-.08	-6.79	6.39	-4.00
		-3.60	-1.39	9.36	3.52	-3.31	5.96	.01	-.07	-.11	-.61	-4.00	16.90
	50	1.08	.23	2.28	1.18	1.03	-1.97	.56	-.08	-.08	.99	1.35	.49
		.23	6.23	1.96	1.03	8.48	-5.15	.04	.27	-.11	1.35	10.73	-1.80
		2.28	1.96	9.27	-1.97	-5.15	6.93	.03	-.06	.05	.49	-1.80	15.45
	32	.75	1.08	.64	1.06	-.70	-1.98	.14	.19	.12	1.57	.06	-1.65
C		1.08	3.68	-1.80	-.70	2.14	.89	-.04	.26	.23	.06	4.37	-1.68
		.64	-1.80	4.79	-1.98	.89	5.98	.02	.01	-.03	-1.65	-1.68	11.09
	27	50.78	-108.04	2.47	59.74	119.72	4.42	.96	-.86	.12	5.24	3.51	2.24
CC		108.04	270.28	.89	119.72	264.70	-11.00	1.06	.99	-.21	3.51	4.18	3.47
		2.47	.89	9.99	4.42	-11.00	2.25	.15	-.12	.11	2.24	3.47	11.23
7/21/88	52	5.46	.27	1.33	2.42	1.17	-.51	.54	.34	-.03	3.93	.76	-.88
MK-27		.27	.59	-1.27	1.17	1.30	.33	.10	.07	-.10	.76	1.77	-1.12
		1.33	-1.27	11.53	-.51	.33	8.84	.27	.25	.51	-.88	-1.12	10.15
	28	1.17	-1.54	1.25	.95	.92	-2.17	.17	.25	-.10	1.77	-.79	-1.15
		-1.54	2.42	-.81	.92	1.66	-.59	-.15	-.33	-.11	-.79	5.42	-1.31
		1.25	-.81	3.27	-2.17	-.59	8.41	.13	.04	-.35	-1.15	-1.31	15.38
	21	.65	-1.23	1.63	.18	.01	-.32	.38	.15	.62	.57	-1.32	.51
		-1.23	6.29	-4.26	.01	2.77	.73	-.04	-.20	-.16	-1.32	10.68	-3.03
		1.63	-4.26	5.06	-.32	.73	2.80	.16	.02	.26	.51	-3.03	5.88
	50	2.77	-.50	-.58	2.26	-1.81	1.83	-.08	.33	-.13	5.43	-2.67	.99
		-.50	.36	-.78	-1.81	1.82	-1.27	.03	-.06	-.01	-2.67	2.29	-1.77
		-.58	-.78	4.50	1.83	-1.27	2.28	.27	-.28	.26	.99	-1.77	5.13
	0	---	---	---	---	---	---	---	---	---	---	---	---
		---	---	---	---	---	---	---	---	---	---	---	---
		---	---	---	---	---	---	---	---	---	---	---	---
	17	.15	-.27	.73	.26	-.05	-.75	.46	-.10	-.22	.22	.44	-.36
		-.27	7.16	-2.43	-.05	2.92	-.24	-.94	.33	.41	.44	7.03	-3.21
		.73	-2.43	4.14	-.75	-.24	2.28	.99	-.22	-.51	-.36	-3.21	9.57
	20	.87	.20	-.63	.74	1.01	-.65	.18	.53	-.27	1.32	.73	-.78
C		.20	.78	.95	1.01	2.61	-1.97	.05	.58	-.16	.73	1.73	-.62
		-.63	.95	3.95	-.65	-1.97	2.87	-.02	.07	-.12	-.78	-.62	7.66

Date	T	Dx			Dy			Rxy			M		
	50	4.19	-2.29	-2.04	2.66	-1.69	.90	.34	-.23	.00	4.61	-3.26	-1.08
		-2.29	1.66	2.19	-1.69	1.49	-1.46	-.09	.06	.09	-3.26	2.97	.46
		-2.04	2.19	4.80	.90	-1.46	2.85	-.02	.01	.14	-1.08	.46	6.60
7/21/88	40	117.59	-5.91	8.41	114.89	-1.28	-12.84	.98	-.34	-.01	5.55	2.32	-1.02
MK-30		-5.91	.67	2.50	-1.28	3.71	-2.88	-.31	.17	.15	2.32	3.85	.95
		8.41	2.50	212.64	-12.84	-2.88	233.43	-.02	-.44	.97	-1.02	.95	14.68
	23	44.27	-3.88	5.53	41.15	4.05	4.59	.98	.29	.50	1.51	-.90	-.35
		-3.88	3.57	-.96	4.05	3.76	2.17	-.19	-.36	.14	-.90	9.96	-.77
		5.53	-.96	3.29	4.59	2.17	10.53	.22	.24	.22	-.35	-.77	11.23
	25	1.21	-2.57	1.78	.27	.53	-.39	.03	-.04	-.14	1.45	-1.96	1.64
C		-2.57	6.27	-3.16	.53	2.30	.54	-.01	-.06	-.03	-1.96	9.02	-1.86
		1.78	-3.16	3.51	-.39	.54	2.18	-.07	-.18	-.20	1.64	-1.86	6.83
	50	2.29	-.21	-1.13	1.99	-2.02	1.36	-.19	.40	-.07	5.09	-2.73	.12
		-.21	.42	-1.06	-2.02	2.63	-.78	.06	.04	-.02	-2.73	2.96	-1.19
		-1.13	-1.06	4.76	1.36	-.78	2.46	.13	-.57	.17	.12	-1.19	6.05
	30	119.29	216.18	5.06	111.95	216.93	6.56	.99	1.03	.32	1.54	.44	-.19
CC		216.18	418.89	1.31	216.93	439.23	16.56	.92	.99	.36	.44	9.76	-5.47
		5.06	1.31	6.37	6.56	16.56	6.16	.12	.09	-.03	-.19	-5.47	12.94
	50	.32	.17	.93	.23	.01	-.62	.25	-.16	.03	.41	.06	.43
		.17	6.96	-3.21	.01	4.05	-1.89	.28	-.08	-.20	.06	11.88	-3.28
		.93	-3.21	5.91	-.62	-1.89	3.45	-.15	-.17	.14	.43	-3.28	8.08
	50	1.76	.77	.34	.49	1.05	-.35	.11	.03	.07	2.04	1.66	-.18
		.77	.90	2.02	1.05	3.50	-2.20	.05	.07	.00	1.66	4.16	-.28
		.34	2.02	8.61	-.35	-2.20	2.29	.01	.05	.06	-.18	-.28	10.36
	50	.88	.97	-1.11	5.41	1.21	-1.95	.13	.00	.00	5.71	2.82	-2.01
		.97	5.98	3.29	1.21	.58	-1.61	-.93	-.16	.59	2.82	7.15	.86
		-1.11	3.29	7.03	-1.95	-1.61	6.03	-.47	-.06	.17	-2.01	.86	10.89
	50	.48	-1.29	1.13	.90	2.00	-2.65	.03	-.03	-.01	1.34	.45	-1.65
		-1.29	6.55	-1.58	2.00	7.22	-4.99	.18	.05	-.17	.45	13.08	-5.32
		1.13	-1.58	3.87	-2.65	-4.99	8.85	.07	-.04	-.11	-1.65	-5.32	14.01
	27	.72	.53	.73	.88	-1.18	-1.05	.18	-.30	.34	1.30	.57	-1.80
		.53	4.13	-4.00	-1.18	9.74	-11.81	-.25	.00	.27	.57	13.83	-17.34
		.73	-4.00	7.96	-1.05	-11.81	23.56	.14	-.08	-.04	-1.80	-17.34	32.58
	24	1.02	-1.68	2.18	3.70	5.55	-5.68	-.08	.09	.16	5.03	3.31	-3.86
		-1.68	3.92	-1.97	5.55	11.56	-4.41	.06	.02	-.05	3.31	15.25	-7.42
		2.18	-1.97	7.12	-5.68	-4.41	14.07	-.13	.20	.30	-3.86	-7.42	15.10
	50	7.31	-1.41	-4.27	3.59	-2.61	1.15	.03	-.03	.11	10.64	-3.53	-3.19
		-1.41	1.19	1.02	-2.61	2.49	-.79	-.10	-.13	.08	-3.53	4.15	-.21
		-4.27	1.02	5.58	1.15	-.79	2.42	-.10	.08	.21	-3.19	-.21	6.50
	37	1.77	2.51	2.26	1.31	-3.38	1.14	.25	-.54	.27	2.31	-.16	2.36
		2.51	4.16	2.20	-3.38	9.70	-4.71	.14	-.29	.15	-.16	17.52	-2.20
		2.26	2.20	4.83	1.14	-4.71	4.80	.12	-.29	.16	2.36	-2.20	8.09

APPENDIX D. AUTOCORRELATIONS

Examples of autocorrelation functions of residuals $R(h)/R(0)$, (see eq. (41)) are presented in Figure D1 through D6. The first two are for the displacement process, eq. (40). The latter four are for the three components of displacement.

Array 2



Array 11

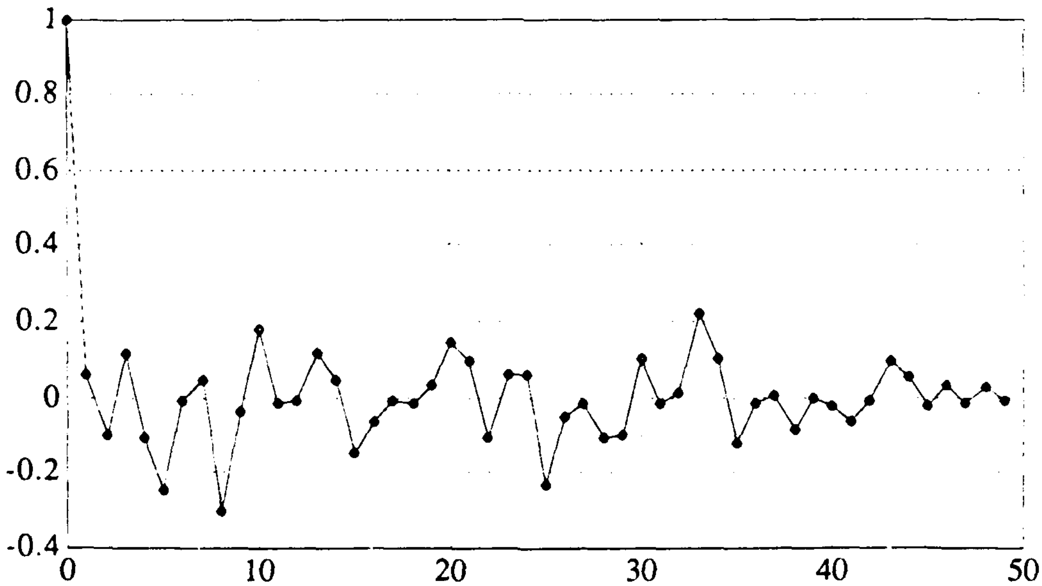
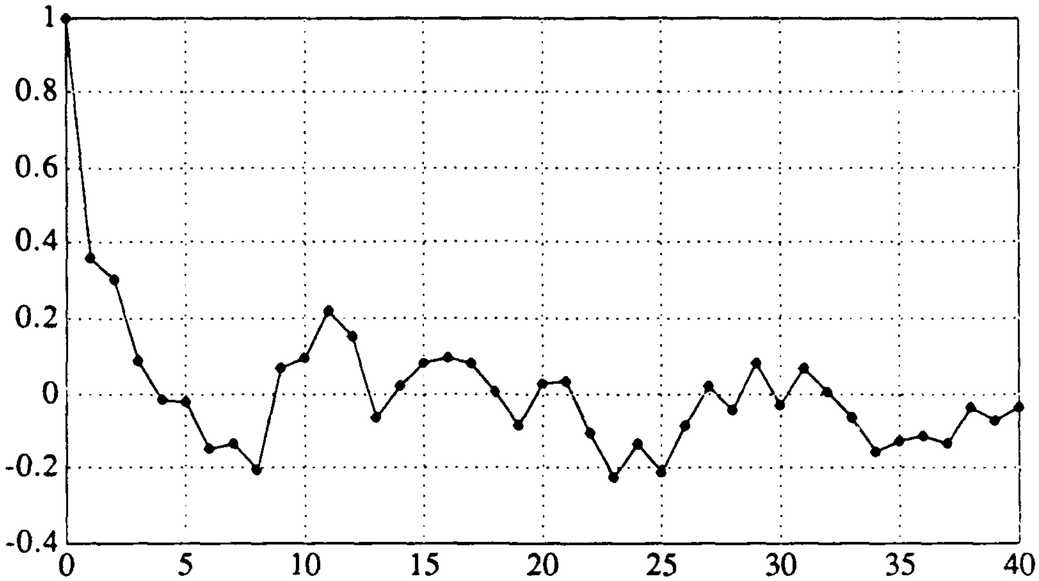


Figure D1. Autocorrelation of noise displacement

3/23/89 $\hat{\delta} = 0.15$

Array 4



Array 54

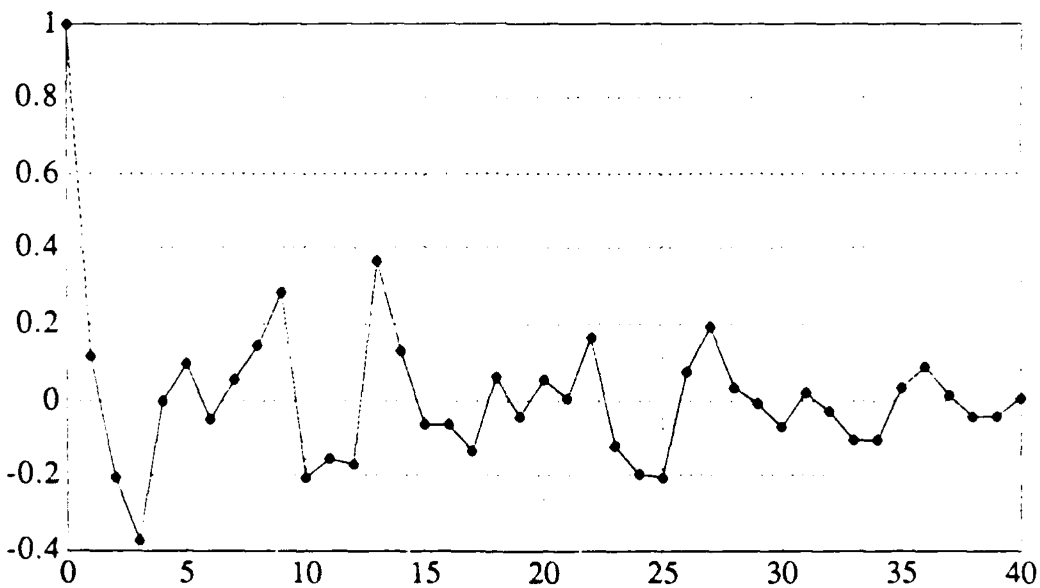


Figure D2. Autocorrelation of noise displacement

7/4/88 $\hat{\delta}$ =1.10

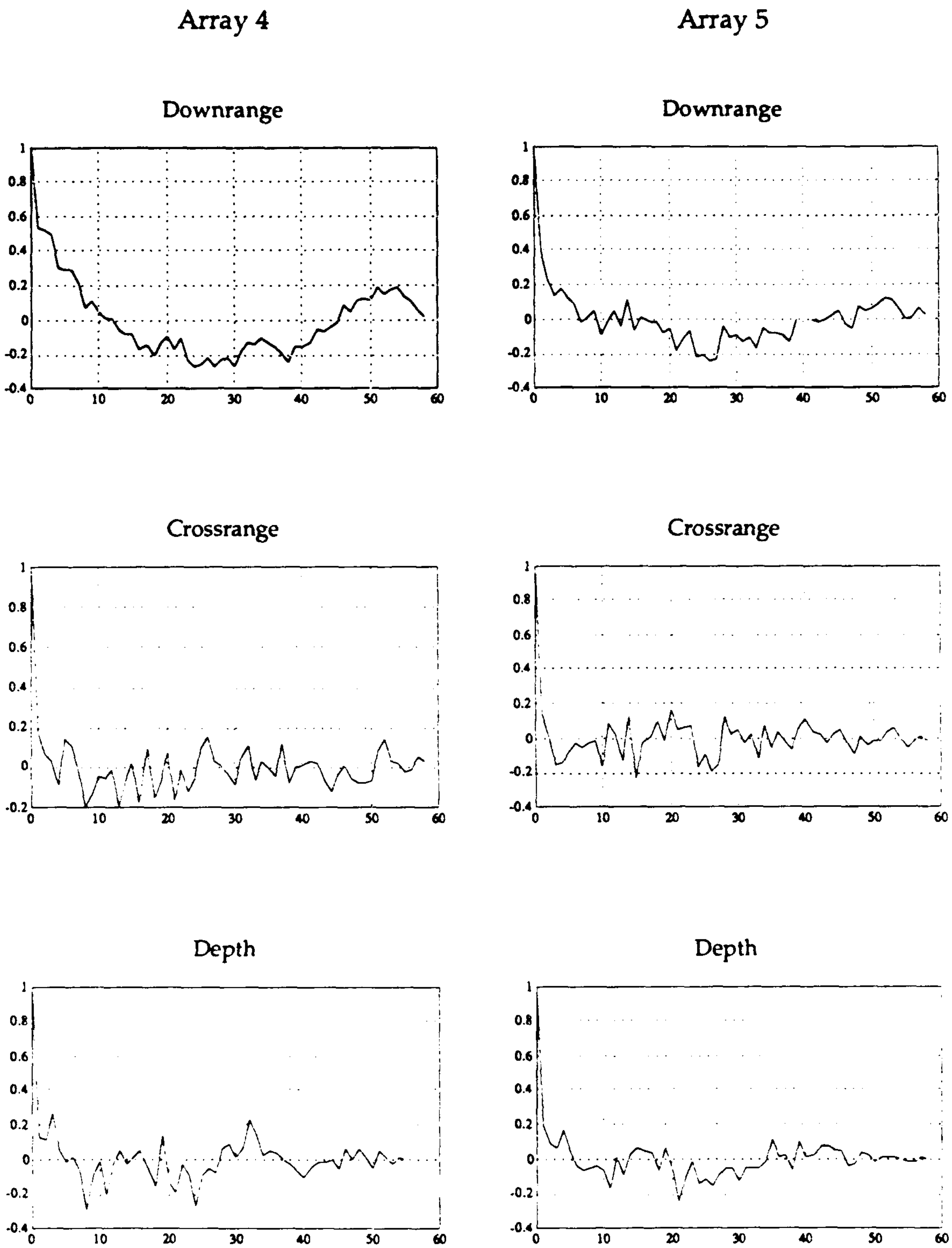


Figure D3. Autocorrelation of noise components

5/10/89 $\hat{\delta} = 1.24$

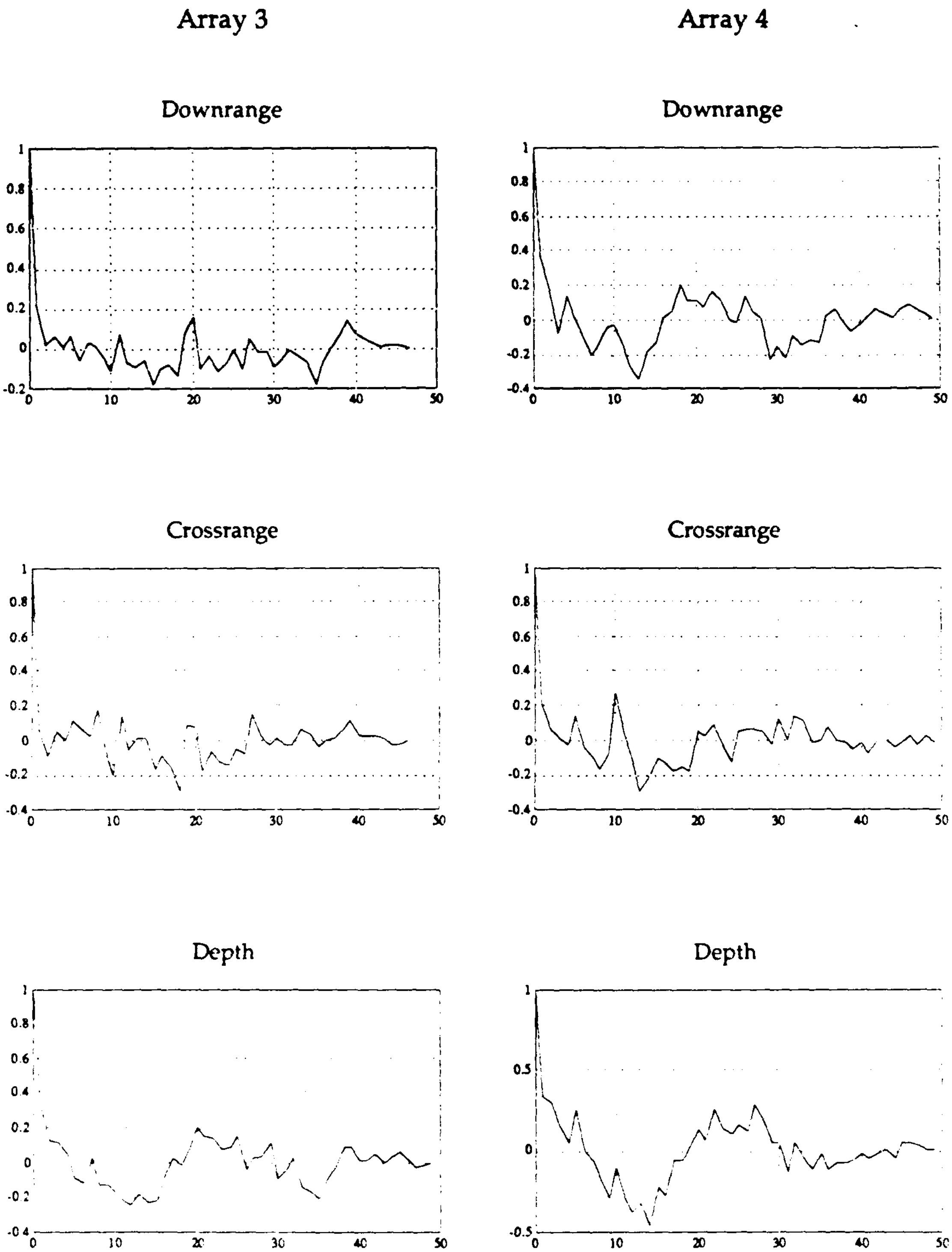


Figure D4. Autocorrelation of noise components

5/10/89 $\hat{\delta} = 2.85$

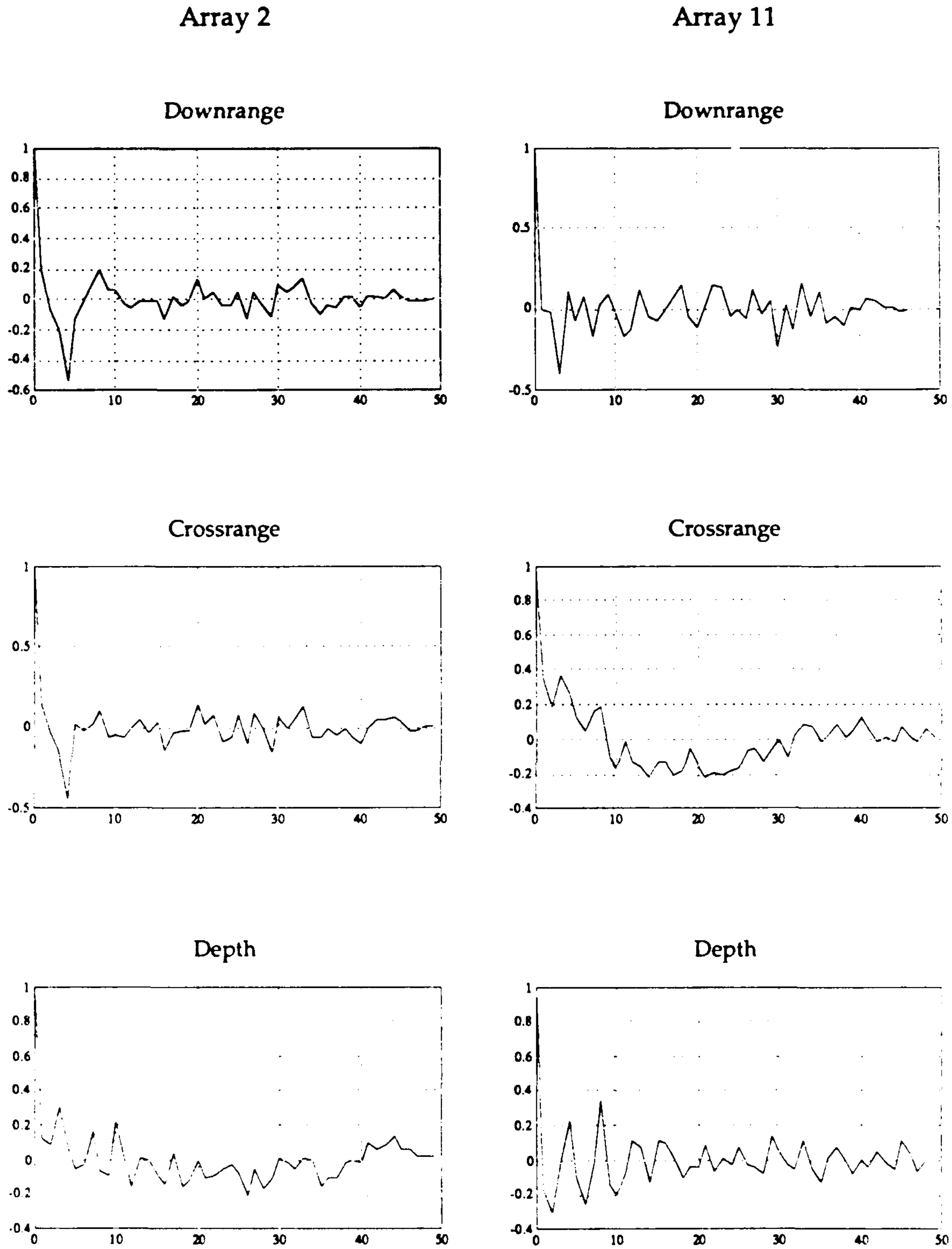


Figure D5. Autocorrelation of noise components

6/06/89 $\hat{\delta} = 2.48$

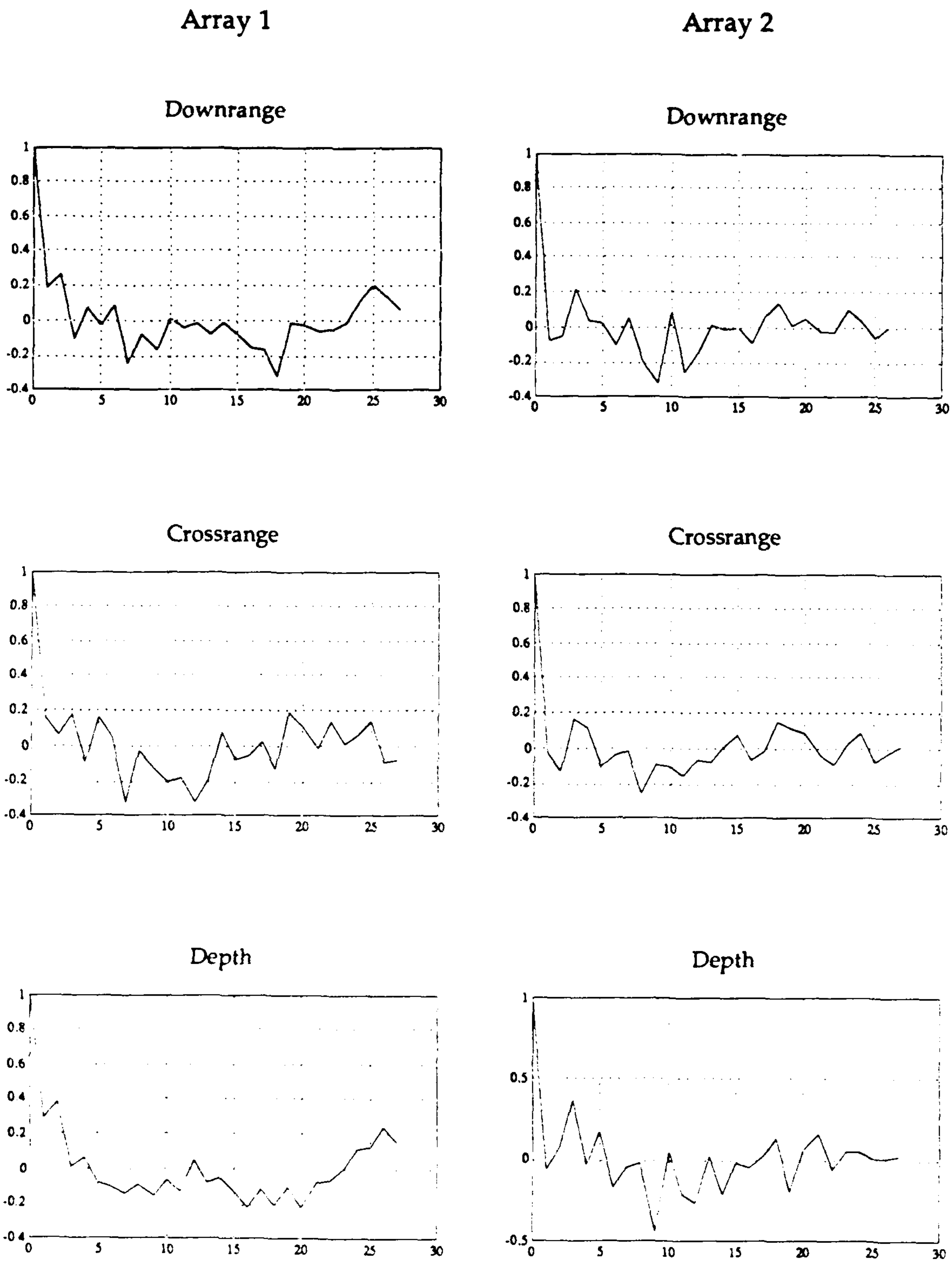


Figure D6. Autocorrelation of noise components

7/21/89 $\hat{\delta} = 2.75$

APPENDIX E. COMPUTER SOURCE CODES

The major programs needed to produce Table 1 are documented here. Basically there are three steps:

(i) Extract pertinent crossover data from T-files

(ii) Produce the covariance matrices of residuals; D_x, D_y, D_{xy} and M. See equations (38) and (39)

(iii) Develop the estimates, eq. (9) and (10). Develop supporting statistics, eq (17), (18), (19)

Also the user needs access to Table E.1, the coordinates of the various arrays at the Nanoose range.

Step (i) is managed by the program KEYGATE, written in FORTRAN 77, Miscosoft optimizing compiler 4.01. This program reads the NUWES T-files and performs a series of gating operations in order to produce crossover data in the required seven column format. It will prompt the user for

(a) name of the range (e.g., Nanoose)

(b) Number of records to ignore (e.g., 31 for bypassing the DVT information)

(c) The target vehicle mode (e.g. 7)

First, it will strip out all data other than mode 2 or mode 7. Second, it identifies all point counts for which a position vector is available for two or more arrays. Third, it reads an array location file and removes data that cannot be reasonably identified with an overlap region. Next, it organizes the data by pairs of arrays. Finally, it prompts the user for his array pair and output filename; it selects and records all the data of the desired type.

Step (ii) is contained in the program PRINCOM3.M. This is a PCMATLAB program that performs the eigen analysis, eq. (32); develops the smoothed fit, eq. (37); and the covariance matrices (38) and (39). The latter is placed on an output file to be used in Step (iii). It also develops the autocorrelation sequence, e.g. (41) of the displacement process.

A slight modification of this program can be made to develop autocorrelations for the three components of residuals instead of for the displacement process.

Step (iii) is performed by the program TIMECOR, a FORTRAN 77 code. The Microsoft FORTRAN optimizing compiler 4.01 is used. The user should note that requirements of the three input files:

VELOCITY.DAT is a two-column data set containing the water layer boundaries and the sound speeds in 25 ft increments

TRPDOTRX.DAT is the seven-column crossover data output of KEYGATE to which has been appended the locations of the two arrays as the last record.

MMATRIX.M is the covariance matrix, M, produced by PRINCOM.M in Step (ii).

Since the ray tracing exit layer elevation angles are not contained in T-files it is necessary to reconstruct them in order to compute Equation (5). This is accomplished by the subroutine TGEN. The methodology of TGEN is explained in [3] under the heading of ray fitting.

There are two output files which, taken together, contain all of the information indicated in Table 1.

TABLE E.1. COORDINATES OF THE NANOOSE ARRAYS

Array Number	Date of Survey	Downrange	Crossrange	Depth
0	6/20/85	12188.01	−131.52	−1295.33
1	6/20/85	19463.16	−174.99	−1308.76
2	7/12/85	26991.39	−109.83	−1323.25
3	1/07/88	34505.10	−80.76	−1323.32
4	10/24/88	42005.19	−55.17	−1318.28
5	6/20/85	49497.00	−25.23	−1315.58
6	6/20/85	56972.28	−21.21	−1308.50
7	7/30/85	64680.66	15.33	−1353.39
8	11/16/88	71969.73	−29.28	−1300.89
9	5/07/84	3.00	3.00	1.00
10	3/12/84	47100.00	−3600.00	−1300.00
11	7/18/85	23173.89	−6488.40	−1312.09
12	6/20/85	30731.25	−6553.05	−1312.90
13	6/20/85	38213.61	−6640.77	−1323.05
14	6/20/85	45647.07	−6513.18	−1324.78
15	6/19/85	53249.43	−6354.60	−1316.66
16	9/13/85	60859.74	−6356.07	−1313.42
17	6/16/87	68217.93	−6524.10	−1313.43
54	2/02/88	38029.95	5401.98	−1212.69
55	6/20/85	45645.75	6369.66	−1188.12
56	7/30/85	53180.13	6417.96	−1218.84
57	7/30/85	60745.71	6419.40	−1088.24
23	6/20/85	41605.14	−12150.18	−1268.23
24	4/17/89	49572.00	−12966.00	−1300.00
25	10/24/88	56993.79	−12999.33	−1205.48
26	8/08/88	64442.94	−12971.04	−1255.35
27	7/15/80	22119.60	−15908.70	83.00
28	5/04/83	45000.00	1500.00	−1350.00
29	2/02/79	.00	.00	.00

KEYGATE

```
PROGRAM KEYGATE
```

> Program to read in raw data from Keyport hydrophone arrays, segregate it by mode, and throw out unusable records. The output of this program is to be read in by the program KEYMAIN.
> Modified by Colin R. Cooper 11/15/89

```
      INTEGER*4 CRT, KBD
      CHARACTER*25 DSNAME, SITNAM
      CHARACTER*9 TEMP1, TEMP2, TEMP3, TEMP4

      PARAMETER (KBD=5, CRT=6)

      WRITE(CRT,*) ' Please enter the name of your input file: '
      READ(KBD,'(A)') DSNAME

      WRITE(CRT,*) ' Please enter the name of the range configuration file:',
      READ(KBD,'(A)') SITNAM

      TEMP1 = 'TEMP1.TMP'
      TEMP2 = 'TEMP2.TMP'
      TEMP3 = 'TEMP3.TMP'
      TEMP4 = 'TEMP4.TMP'

      CALL STRMOD(CRT,KBD,TEMP1,DSNAME)

      CALL PAIR(CRT,KBD,TEMP1,TEMP2)

      CALL RANGE(CRT,KBD,SITNAM,TEMP2,TEMP3)

      CALL PAIR2(CRT,KBD,TEMP3,TEMP4)

      CALL REC(CRT,KBD,TEMP4)

      WRITE(CRT,*)' Operation complete.  KEYGATE terminating... '

      STOP
      END


      SUBROUTINE STRMOD(CRT,KBD,TEMP1,DSNAME)
```

> Program to strip all modes except 2's and 7's from the Keyport data. (2 indicates target ship, 7 indicates torpedo).

```
      CHARACTER DO*2, DSNAME*25, TEMP1*9
      INTEGER PC, ARRAY, CRT, NHEAD

      OPEN(1,FILE=DSNAME,STATUS='OLD')

      WRITE(CRT,*)' How many records of header do you want to strip off the file?',
      READ(KBD,*) NHEAD
      DO 11 I = 1,NHEAD
        READ(1,*)
 11   CONTINUE
```

```
      WRITE(CRT,*)' Input mode to be kept?'
      READ(KBD,*) NUM

      OPEN(2,FILE=TEMP1,STATUS='NEW')

 10   READ(1,100,END=50,ERR=40)PC,DO,X,Y,Z,ARRAY,MODE
      IF(DO .NE. '  ')GOTO 20
      IF(MODE .NE. NUM) GOTO 20
      WRITE(2,110)PC,X,Y,Z,ARRAY,MODE
 20   CONTINUE
      GOTO 10
 40   WRITE(CRT,*)' THERE WAS AN ERROR IN THE FILE'
 50   CONTINUE
100   FORMAT(I5,A2,1X,F7.1,2X,F7.1,2X,F7.1,30X,I2,2X,I1)
110   FORMAT(1X,I5,2X,3F10.1,2X,I2,2X,I2)
      CLOSE (UNIT=1)
      CLOSE(UNIT=2)
      RETURN
      END


      SUBROUTINE PAIR(CRT,KBD,TEMP1,TEMP2)
```

Program to pair point counst after the data has been gated by STRMOD. Second pass.

```
      DIMENSION X(200), Y(200), Z(200)
      INTEGER*4 PC(200), ARRAY(200), MODE(200), CRT, HOLD
      CHARACTER*9 TEMP1, TEMP2

      OPEN(1,FILE=TEMP1,STATUS='OLD')
      OPEN(2,FILE=TEMP2,STATUS='NEW')

      HOLD = 0
      IREC = 0
      NREC = 0
      I = 1
```

Read records by two's to compare point counts.

```
      READ(1,*,END=40,ERR=30) PC(I), X(I), Y(I), Z(I),
     +        ARRAY(I), MODE(I)
 10   READ(1,*,END=40,ERR=30) PC(I+1), X(I+1), Y(I+1), Z(I+1),
     +        ARRAY(I+1), MODE(I+1)
      NREC = NREC + 1

      IF(PC(I+1) .EQ. HOLD) THEN
         WRITE(2,100) PC(I+1), X(I+1), Y(I+1), Z(I+1),
     +          ARRAY(I+1), MODE(I+1)
         HOLD = PC(I+1)
         GO TO 20
      END IF

      IF(PC(I) .EQ. PC(I+1)) THEN
         WRITE(2,100) PC(I), X(I), Y(I), Z(I), ARRAY(I), MODE(I)
         WRITE(2,100) PC(I+1), X(I+1), Y(I+1), Z(I+1),
```

```
     +              ARRAY(I+1), MODE(I+1)
           HOLD = PC(I+1)
           IREC = IREC + 1
           GO TO 20
         END IF

         IF(PC(I) .NE. PC(I+1)) THEN
           PC(1) = PC(I+1)
           X(1) = X(I+1)
           Y(1) = Y(I+1)
           Z(1) = Z(I+1)
           ARRAY(1) = ARRAY(I+1)
           MODE(1) = MODE(I+1)
           I = 1
         END IF

         GOTO 10
   20    I = I + 1
         GO TO 10

   30    WRITE(CRT,*)' THERE IS AN ERROR IN THE DATA FILE IN REC.',NRE
         WRITE(CRT,*)' ... OPERATION TERMINATING DUE TO ERROR. '
         STOP
   40    CONTINUE

         CLOSE(UNIT=1)
         CLOSE(UNIT=2)

  100    FORMAT(1X,I5,2X,F7.1,2X,2F9.1,2X,I2,2X,I2)

         RETURN
         END


         SUBROUTINE  RANGE(CRT,KBD,SITNAM,TEMP2,TEMP3)
```

This program completes the third gating of Keyport range data. It reads array location data from a site specific configuration file and tests to see if the data is in the valid overlap area.

```
         INTEGER*4 ARRAY, CARRAY, CRT, P
         REAL*4  CONFIG(200,4), LX, LY, LZ, MAXVAL
         CHARACTER SITNAM*25, TEMP2*9, TEMP3*9
```

Open input and output files:

```
         OPEN(1,FILE=TEMP2,STATUS='OLD')
         OPEN(2,FILE=SITNAM,STATUS='OLD')
         OPEN(3,FILE=TEMP3,STATUS='NEW')
```

Read site configuration into CONFIG array:

```
         NREC = 0
         I = 1
   10    READ(2,*,END=40,ERR=30) CONFIG(I,1), CONFIG(I,2), CONFIG(I,3),
```

```
+           CONFIG(I,4)
      NREC = NREC + 1
      I = I + 1
      GOTO 10
30    WRITE(CRT,*)' There was an error reading the config file in record',NRB
40    CONTINUE
      CLOSE(UNIT=2)

      NDREC = 0
```

Read X, Y, Z, and ARRAY from input data file:

```
45    READ(1,*,ERR=70,END=80) PC, X, Y, Z, ARRAY, MODE
      DO 50 I = 1,NRE
         LX = CONFIG(I,1)
         LY = CONFIG(I,2)
         LZ = CONFIG(I,3)
         CARRAY = INT(CONFIG(I,4))
         MAXVAL = 4700.
```

Match array number in data file with that in config. file. If they are equal compute slant range distance (SRDIST):

```
         IF(ARRAY .EQ. CARRAY) THEN
            SRDIST = SQRT( (X - LX)**2 + (Y - LY)**2 + (Z - LZ)**2 )
            IF(MAXVAL .GE. SRDIST) THEN
               WRITE(3,100) PC, X, Y, Z, ARRAY, MODE, SRDIST
               NDREC = NDREC + 1
            END IF
         END IF

50    CONTINUE
      GOTO 45

70    WRITE(CRT,*)' There is an error in the data file. '
80    CONTINUE

      CLOSE(UNIT=1)
      CLOSE(UNIT=3)

100   FORMAT(3X,I5,3(3X,F10.1),2(3X,I2),3X,F8.2)

      RETURN
      END



      SUBROUTINE PAIR2(CRT,KBD,TEMP3,TEMP4)
```

Program to pair point counts after the data has been tested by RANGE. Fourth pass.

```
      DIMENSION X(200), Y(200), Z(200), SRDIST(200)
      INTEGER*4 PC(200), ARRAY(200), MODE(200), CRT, HOLD
      CHARACTER*9 TEMP3, TEMP4
```

```
      OPEN(1,FILE=TEMP3,STATUS='OLD')
      OPEN(2,FILE=TEMP4,STATUS='NEW')

      HOLD = 0
      IREC = 0
      NREC = 0
      I = 1
```

Read records by two's to compare point counts.

```
      READ(1,*,END=40,ERR=30) PC(I), X(I), Y(I), Z(I),
     +        ARRAY(I), MODE(I), SRDIST(I)
   10 READ(1,*,END=40,ERR=30) PC(I+1), X(I+1), Y(I+1), Z(I+1),
     +        ARRAY(I+1), MODE(I+1), SRDIST(I+1)
      NREC = NREC + 1

      IF(PC(I+1) .EQ. HOLD) THEN
         WRITE(2,100) PC(I+1), X(I+1), Y(I+1), Z(I+1),
     +        ARRAY(I+1), MODE(I+1), SRDIST(I+1)
         HOLD = PC(I+1)
         GO TO 20
      END IF

      IF(PC(I) .EQ. PC(I+1)) THEN
         WRITE(2,100) PC(I), X(I), Y(I), Z(I), ARRAY(I), MODE(I),
     +            RDIST(I)
         WRITE(2,100) PC(I+1), X(I+1), Y(I+1), Z(I+1),
     +        ARRAY(I+1), MODE(I+1), SRDIST(I+1)
         HOLD = PC(I+1)
         IREC = IREC + 1
         GO TO 20
      END IF

      IF(PC(I) .NE. PC(I+1)) THEN
         PC(1) = PC(I+1)
         X(1) = X(I+1)
         Y(1) = Y(I+1)
         Z(1) = Z(I+1)
         ARRAY(1) = ARRAY(I+1)
         MODE(1) = MODE(I+1)
         SRDIST(1) = SRDIST(I+1)
         I = 1
      END IF

      GOTO 10
   20 I = I + 1
      GO TO 10

   30 WRITE(CRT,*)' THERE IS AN ERROR IN THE DATA FILE IN REC.',NRE
      WRITE(CRT,*)' ... OPERATION TERMINATING DUE TO ERROR. '
      STOP
   40 CONTINUE

      CLOSE(UNIT=1)
      CLOSE(UNIT=2)

  100 FORMAT(1X,I5,2X,F7.1,2X,2F9.1,2X,I2,2X,I2,2X,F8.2)
```

```
      RETURN
      END

      SUBROUTINE REC(CRT,KBD,TEMP4)
```

Program to produce the final gating of Keyport hydrophone array test data.

```
      INTEGER PC, ARRAY, PC1, PC2, A1, A2, ARRAY1, ARRAY2, CRT
      INTEGER PCA(10), ARRAYA(10), HOLD
      DIMENSION XA(10), YA(10), ZA(10), SRDISTA(10)
      CHARACTER   OUTFIL*25, ANS*1, TEMP4*9, TEMP1*9,RANGE*7

      RANGE='NANOOSE'

   99 WRITE(CRT,*)' What is the name you wish to give to ',
     +        ' your output file? '
      READ(CRT,'(A)') OUTFIL

      WRITE(CRT,*)' Enter the numbers of the arrays to be paired: '
      READ(KBD,*) A1, A2

      OPEN(1,FILE=TEMP4,STATUS='OLD')
      OPEN(2,FILE='TEMP1.DAT',STATUS='OLD')

    5 READ(1,*,END=8,ERR=8) PC, X, Y, Z, ARRAY, MODE, SRDIST
      IF((ARRAY .EQ. A1).OR.(ARRAY .EQ. A2)) THEN
         WRITE(2,100) PC, X, Y, Z, ARRAY,SRDIST
      END IF
      GOTO 5

    8 CONTINUE
      CLOSE (UNIT=1)
```

Routine to pair data again:

```
      REWIND 2
      OPEN(4,FILE='TEMP2.DAT',STATUS='OLD')

      I = 1
      IFLAG = 0
      FIRST = 1
      HOLD = 0
      M = 0

    9 READ(2,100,END=40) PCA(I), XA(I), YA(I), ZA(I),
     +        ARRAYA(I), SRDISTA(I)
      IF(FIRST .EQ. 1) THEN
         FIRST = 0
         HOLD = PCA(I)
      END IF

      IF(PCA(I) .EQ. HOLD) THEN
         I = I + 1
         M = M + 1
         GOTO 9
```

ELSE

In cases where there are three or more reports for a given point count, segregate by comparing SRDIST. if there are more than 3 reports, discard all.

```
        IF (M .EQ. 3) THEN
          IF(ARRAYA(M) .EQ. ARRAYA(M-1)) THEN
            IF (ARRAYA(M) .EQ. ARRAYA(M-2)) THEN
            GOTO 50
          ELSE
            IF (ABS(SRDISTA(M-2)-SRDISTA(M)) .LT.
     +           ABS(SRDISTA(M-2)-SRDISTA(M-1))) THEN
              WRITE(4,100) PCA(M), XA(M), YA(M), ZA(M),
     +           ARRAYA(M), SRDISTA(M)
              WRITE(4,100) PCA(M-2), XA(M-2), YA(M-2), ZA(M-2),
     +           ARRAYA(M-2), SRDISTA(M-2)
              IFLAG = 1
            ELSE
              WRITE(4,100) PCA(M-1), XA(M-1), YA(M-1), ZA(M-1),
     +           ARRAYA(M-1), SRDISTA(M-1)
              WRITE(4,100) PCA(M-2), XA(M-2), YA(M-2), ZA(M-2),
     +           ARRAYA(M-2), SRDISTA(M-2)
              IFLAG = 1
            END IF
          END IF
        ELSE
          IF (ARRAYA(M) .EQ. ARRAYA(M-2)) THEN
            IF (ABS(SRDISTA(M-1)-SRDISTA(M)) .LT.
     +           ABS(SRDISTA(M-1)-SRDISTA(M-2))) THEN
              WRITE(4,100) PCA(M), XA(M), YA(M), ZA(M),
     +           ARRAYA(M), SRDISTA(M)
              WRITE(4,100) PCA(M-1), XA(M-1), YA(M-1), ZA(M-1),
     +           ARRAYA(M-1), SRDISTA(M-1)
              IFLAG = 1
            ELSE
              WRITE(4,100) PCA(M-1), XA(M-1), YA(M-1), ZA(M-1),
     +           ARRAYA(M-1), SRDISTA(M-1)
              WRITE(4,100) PCA(M-2), XA(M-2), YA(M-2), ZA(M-2),
     +           ARRAYA(M-2), SRDISTA(M-2)
              IFLAG = 1
            END IF
          ELSE
            IF (ABS(SRDISTA(M)-SRDISTA(M-1)) .LT.
     +           ABS(SRDISTA(M)-SRDISTA(M-2))) THEN
              WRITE(4,100) PCA(M), XA(M), YA(M), ZA(M),
     +           ARRAYA(M), SRDISTA(M)
              WRITE(4,100) PCA(M-1), XA(M-1), YA(M-1), ZA(M-1),
     +           ARRAYA(M-1), SRDISTA(M-1)
              IFLAG = 1
            ELSE
              WRITE(4,100) PCA(M), XA(M), YA(M), ZA(M),
     +           ARRAYA(M), SRDISTA(M)
              WRITE(4,100) PCA(M-2), XA(M-2), YA(M-2), ZA(M-2),
     +           ARRAYA(M-2), SRDISTA(M-2)
              IFLAG = 1
            END IF
          END IF
```

```
        END IF


        ELSE
           IF ((M .EQ. 2) .AND. (ARRAYA(M) .NE. ARRAYA(M-1))) THEN
              WRITE(4,100) PCA(M), XA(M), YA(M), ZA(M),ARRAYA(M),
     +                 SRDISTA(M)
              WRITE(4,100) PCA(M-1), XA(M-1), YA(M-1), ZA(M-1),
     +                 ARRAYA(M-1), SRDISTA(M-1)

           END IF
        END IF
 50     CONTINUE

 25     CONTINUE

        IFLAG = 0
        PCA(1) = PCA(I)
        XA(1) = XA(I)
        YA(1) = YA(I)
        ZA(1) = ZA(I)
        ARRAYA(1) = ARRAYA(I)
        SRDISTA(1) = SRDISTA(I)
        I = 2
        M = 1
        HOLD = PCA(1)

        DO 45 L = 2,10
           PCA(L) = 0
           XA(L) = 0
           YA(L) = 0
           ZA(L) = 0
           ARRAYA(L) = 0
           SRDISTA(L) = 0
 45     CONTINUE

      END IF

      GOTO 9
 40   CONTINUE

      NREC = 0

      CLOSE (UNIT=4)

      OPEN(7,FILE='TEMP2.DAT',STATUS='OLD')
      OPEN(9,FILE=OUTFIL,STATUS='NEW')

      WRITE(9,300) RANGE, A1, A2
```

Read in array data in two record pairs:

```
 10   READ(7,100,END=60,ERR=70) PC1, X1, Y1, Z1, ARRAY1, SRDIST1
      READ(7,100,END=60,ERR=70) PC2, X2, Y2, Z2, ARRAY2, SRDIST2

      IF(ARRAY1 .EQ. A1 .AND. ARRAY2 .EQ. A2) THEN
```

If arrays are in specified order (e.g. 4,5):

```
      WRITE(9,200) PC1, X1, Y1, Z1, X2, Y2, Z2
      END IF

      IF(ARRAY1 .EQ. A2 .AND. ARRAY2 .EQ. A1) THEN
```

If arrays are in reverse order (eg. 5,4):

```
      WRITE(9,200) PC1, X2, Y2, Z2, X1, Y1, Z1

      END IF
```

Increment record counter:

```
      NREC = NREC + 1

      GOTO 10

70    WRITE(CRT,*)' There is a bad record in the file.'
60    CONTINUE

      CLOSE(UNIT=2)
      CLOSE(UNIT=7)
      CLOSE(UNIT=9)

100   FORMAT(I5,3(2X,F8.1),2X,I2,2X,F8.2)
200   FORMAT(2X,I5,1X,6(2X,F11.1))
300   FORMAT(16X,A10,2X,I2,3X,I2)

      WRITE(CRT,*) ' Do you want to try another array pair? (Y/N)'
      READ(KBD,'(A)') ANS
      IF(ANS .EQ. 'Y' .OR.  ANS .EQ. 'y') GO TO 99

      RETURN
      END
```

PRINCOM3

```
function[Autocorx,Autocory]=princom3(fname)
```

3/13/90

PRINCOM3.M will evaluate the principal components of the passed data file. Computes the autocorrelation of the displacement process.

```
iname = ['d:\mdata\' fname '.out'];

eval(['load ' iname ]);
eval(['keyout = ' fname ';']);
eval([ clear ' fname]);
```

```
X=keyout(:,5:7);
Y=keyout(:,2:4);

[len,n]=size(keyout);
```

Compute averages and covariance matices for the tracks produced by each of the arrays.

```
Xav=ones(len,1)*sum(X)/len          ;
Yav=ones(len,1)*sum(Y)/len          ;

XX=X-Xav ;
YY=Y-Yav ;

ipc=keyout(:,1);

CX=cov(XX);
CY=cov(YY);
```

Develop the eigen analysis. Choose largest eigen value.

```
[WX,DX]=eig(CX)     ;
[WY,DY]=eig(CY)     ;

if DX(1,1)>DX(2,2)
        if DX(1,1)>DX(3,3)
            lx=1 ;
        end
elseif DX(2,2)>DX(3,3)
        lx=2    ;
end
if DX(3,3)>DX(1,1)
        if DX(3,3)>DX(2,2)
            lx=3 ;
        end
end
if DY(1,1)>DY(2,2)
        if DY(1,1)>DY(3,3)
            ly=1 ;
        end
elseif DY(2,2)>DY(3,3)
        ly=2    ;
end
if DY(3,3)>DY(1,1)
        if DY(3,3)>DY(2,2)
            ly=3 ;
        end
end

PX=XX*WX;
PY=YY*WY;

ux=PX(:,lx);
uy=PY(:,ly);
```

Modify data projection onto the first principal component to account for constant speed.

```
ipcav=sum(ipc)/length(ipc);
```

```
mx=(sum(ux.*(ipc-ipcav)))/(sum((ipc-ipcav).*(ipc-ipcav))) ;
my=(sum(uy.*(ipc-ipcav)))/(sum((ipc-ipcav).*(ipc-ipcav))) ;

ax=(sum(ux)/length(ux)) - (mx*ipcav);
ay=(sum(uy)/length(uy)) - (my*ipcav);

uhx=(mx*ipc)+ax;
uhy=(my*ipc)+ay;

PPX=zeros(len,3);
PPY=zeros(len,3);

PPX(:,lx)=uhx;
PPY(:,ly)=uhy;
```

Compute the straight line tracks in the original coordinate system.

```
XSL=(PPX*WX')+Xav;
YSL=(PPY*WY')+Yav;
```

Develop the displacement process of residuals.

```
distx = sqrt((X(:,1) - XSL(:,1)).^2 + (X(:,2) - XSL(:,2)).^2 + (X(:,3) - XSL(:,3)).^2);
disty = sqrt((Y(:,1) - YSL(:,1)).^2 + (Y(:,2) - YSL(:,2)).^2 + (Y(:,3) - YSL(:,3)).^2);

davx = sum(distx)/length(distx);
davy = sum(disty)/length(disty);

vx = distx - davx;
vy = disty - davy;
```

Compute the auto correlations.

```
Autocorx = zeros(length(vx),1);
Autocory = zeros(length(vy),1);

for k = 1:length(vy)
      for i = 1:length(vy) - k + 1
         Autocorx(k) = Autocorx(k) + vx(i)*vx(i + k-1);
         Autocory(k) = Autocory(k) + vy(i)*vy(i + k-1);
      end
      Autocorx(k) = Autocorx(k)/sum(vx.^2);
      Autocory(k) = Autocory(k)/sum(vy.^2);
end

clg
subplot(211)
plot(Y(:,1),Y(:,2),'or',YSL(:,1),YSL(:,2),'-r',X(:,1),X(:,2),'xg',XSL(:,1),XSL(:,2),'-g')
title([fname ' - Corrected Tracks with Principal Components'])
xlabel('Down Range'),ylabel('Cross Range')
subplot(212)
plot(Y(:,1),Y(:,3),'or',YSL(:,1),YSL(:,3),'-r',X(:,1),X(:,3),'xg', XSL(:,1),XSL(:,3),'-g')
title('Corrected Tracks with Principal Components')
xlabel('Down Range'),ylabel('Depth')
```

```
pause
clg
h = 0:length(Autocorx)-1;
subplot(211),plot(h,Autocory,'or',h,Autocory,'-r')
title([fname ' - Autocorrelation of Distance, Y Array']),grid
subplot(212),plot(h,Autocorx,'or',h,Autocorx,'-r')
title('Autocorrelation of Distance, X Array'),grid
clg
plot(h(1:27),Autocorx(1:27),'or',h(1:27),Autocorx(1:27),'-r')
title('Autocorrelation of Residual Distance'),grid
xlabel('Point Count Lag')
```

TIMCOR

PROGRAM TIMECOR

06/06/90

This program estimates the timing synchronization offset and drift parameters based upon cross-over data from T-files. Since exit angles and transit times are not recorded to these files they must be re-estimated. Location information for each of the involved arrays is also required.

This program was compiled using:
Microsoft FORTRAN OPTIMIZING COMPILER ver. 4.01

This files must be able to access the following files

VELOCITY.DAT	- Sound velocity vs. depth data.
TRPDOTRX.DAT	- Torpedo tracking data.
MMATRIX.M	- Correlation matrices for data.
OUTPUT1.COV	- Output file containing delta and tnot
OUTPUT2.COV	- Output file containing cross correlation and M matrices.

Users Notes:

- VELOCITY.DAT file contains layer boundaries and sound velocities for 25 ft. depth increments.
- TRPDOTRX.DAT file contains point counts and three position components for each of the two contributing arrays. The position data from the lower numbered array is columns two through four. The last record contains the coordinate position of the two arrays..
- MMATRIX.M file contains the set of covariance matrices produced by AUTOCOR3.M.
- The two output files contain all the information appearing in Table 1.

```
DIMENSION IPC(150)
CHARACTER*30 LINE
REAL*8 Y1(150,2),Y2(150,2),Y3(150,2),YC1(150,2),YC2(150,2)
REAL*8 LL(55),G(55),VV(55),A2,P1,P2,V0,V1,TTIME(150,2)
REAL*8 THETA(150,2),DEPTH(55),PHI(150,2),V(150,2),DZ
REAL*8 B1(150,2),B2(150,2),B3(150,2),YC3(150,2)
REAL*8 DEN,GG,SSB,SSR,TD,TM,SIG,DD,TEMP
REAL*8 MEQ,GEQ(150),DELTA,T1,TNOT,D1,D2,MSR,MSB
REAL*8 DUM1,DUM2,DUM3,DUM4,DUM5,DUM6,DUM7
REAL*8 M(3,3),C(150,3),VDEL,SDEL,VM,SDVM,COVDM
```

```
      REAL*8 DELTAN,MEQN,R,CX(3,3),CY(3,3),CXY(3,3),RXY(3,3),T
      INTEGER*4 TEE
```

Read in the data from the data files.

```
      OPEN(UNIT=2,FILE='VELOCITY.DAT',STATUS='OLD')
      OPEN(UNIT=7,FILE='TRPDOTRX.DAT',STATUS='OLD')
      OPEN(UNIT=10,FILE='MMATRIX.M',STATUS='OLD')
      OPEN(UNIT=11,FILE='OUTPUT1.COV',STATUS='OLD')
      OPEN(UNIT=12,FILE='OUTPUT2.COV',STATUS='OLD')

      READ(10,120)CX(1,1),CX(2,1),CX(3,1),CX(1,2),CX(2,2),
     *   CX(3,2),CX(1,3),CX(2,3),CX(3,3)
      READ(10,120)CY(1,1),CY(2,1),CY(3,1),CY(1,2),CY(2,2),
     *   CY(3,2),CY(1,3),CY(2,3),CY(3,3)
      READ(10,120)CXY(1,1),CXY(2,1),CXY(3,1),CXY(1,2),CXY(2,2),
     *   CXY(3,2),CXY(1,3),CXY(2,3),CXY(3,3)
      READ(10,124)T
      TEE = IDINT(T)

      IF (TEE.EQ.0) THEN
         DO 2 I = 1,3
            RXY(1,I) = 0.0D0
            RXY(2,I) = 0.0D0
            RXY(3,I) = 0.0D0
 2       CONTINUE
      ELSE
         DO 4 I = 1,3
            DO 3 J = 1,3
               RXY(J,I)=CXY(J,I)/DSQRT(CX(I,I)*CY(J,J))
               M(J,I) = CX(J,I) + CY(J,I) - CXY(I,J) - CXY(J,I)
 3          CONTINUE
 4       CONTINUE
      ENDIF

 5    READ(12,'(A)',END=7)LINE
      GOTO 5
 7    BACKSPACE 12
      WRITE(12,122)TEE
      DO 8 I = 1,3
 8       WRITE(12,121)CX(I,1),CX(I,2),CX(I,3),CY(I,1),CY(I,2),
     *         CY(I,3),RXY(I,1),RXY(I,2),RXY(I,3),M(I,1),M(I,2),M(I,3)
      WRITE(12,123)

      I=1
 10   READ(7,*)IPC(I),Y1(I,1),Y2(I,1),Y3(I,1),
      IF(IPC(I).EQ.999) GOTO 15
      I=I+1
      GOTO 10
 15   LEN=I-1

 120  FORMAT(1X,9(E15.4,1X))
```

```
121  FORMAT(5X,I2,5X,'│',25X,'│',2X,'│',25X,'│',2X,'│',25X,'│'
  *      2X,'│',25X,'│')
122  FORMAT(12X,'┌',1X,3(F7.2,1X),'┐',2X,'┌',1X,3(F7.2,1X),'┐',
  *      2X,'┌',1X,3(F7.2,1X),'┐',2X,'┌',1X,3(F7.2,1X),'┐')
123  FORMAT(12X,'└',25X,'┘',2X,'└',25X,'┘',2X,'└',25X,'┘'
  *      2X,'└',25X,'┘')
124  FORMAT(1X,E15.4)
```

Read the VELOCITY.DAT file and prepare for isogradient raytracing.

```
     DZ=25
     I = 1
 25  READ(2,*,END=30) TEMP,VV(I)
     I = I+1
     GOTO 25
 30  CONTINUE
     GG=(VV(I-1)-VV(I-7))/6.
     DO 35 J = I,55
        VV(J)=VV(J-1)+GG
 35  CONTINUE
     LL(1)=12.5
     DEPTH(1)=0
     DO 40 I=2,55
        LL(I)=LL(I-1)+DZ
        DEPTH(I)=DEPTH(I-1)+DZ
 40  CONTINUE
     DO 43 I=2,55
        G(I-1)=(VV(I)-VV(I-1))/DZ
 43  CONTINUE
     G(55)=G(54)+GG/DZ
     DO 45 I=1,52
 45  CONTINUE
     DO 80 J=1,2
        N=15
        IFLAG=0
        DO 70 IT=1,LEN
```

Set variables the the call to TGEN subroutine. TGEN returns the time and elevation.

```
           A2=Y3(LEN+1,J)
           P1=SQRT((Y1(IT,J)-Y1(LEN+1,J))**2+(Y2(IT,J)-Y2(LEN+1,J))**2)
           P2=-Y3(IT,J)
           IF(IT.NE.1) THETA(IT,J)=THETA(IT-1,J)
           CALL TGEN(LL,G,VV,A2,P1,P2,THETA(IT,J),TTIME(IT,J),
  *              V0,V1,DEPTH,IFLAG)
           IFLAG=1
```

Find the azimuth angles from each array.

```
           PHI(IT,J)=DASIN((Y2(IT,J)-Y2(LEN+1,J))/SQRT((Y2(IT,J)-
```

```
     *          Y2(LEN+1,J))**2 + (Y1(IT,J)-Y1(LEN+1,J))**2))
            IF((Y1(IT,J)-Y1(LEN+1,J)).LT.0) PHI(IT,J)=3.14159265359
     *        -PHI(IT,J)
```

Locate the layer containing the source and set it's velocity.

```
  50        IF(-Y3(IT,J).LE.DEPTH(N).AND.-Y3(IT,J).GT.DEPTH(N-1)) GOTO 60
            IF(-Y3(IT,J).GT.DEPTH(N)) THEN
     *        N=N+1
            ELSE
              N=N-1
              IF(N.LE.1) THEN
                N=1
                GOTO 60
              ENDIF
            ENDIF
            GOTO 50
  60        V(IT,J)=VV(N)
```

Calculate the B(T) adjust...ents (Sperical Coordinates : Equtaion (5))

```
            B1(IT,J)=V(IT,J)*DCOS(THETA(IT,J))*DCOS(PHI(IT,J))
            B2(IT,J)=V(IT,J)*DCOS(THETA(IT,J))*DSIN(PHI(IT,J))
            B3(IT,J)=V(IT,J)*DSIN(THETA(IT,J))
  70      CONTINUF
  80    CONTINUE
        DO 90 I=1,LEN
  90    CONTINUE
```

Calculate DELTA and TNOT.

```
        D2=0
        T1=0
        DEN=0
        DO 200 IT=1,LEN
          D1=(((B1(IT,1)-B1(IT,2))*(B1(IT,1)-B1(IT,2)))+
     *        ((B2(IT,1)-B2(IT,2))*(B2(IT,1)-B2(IT,2)))+
     *        ((B3(IT,1)-B3(IT,2))*(B3(IT,1)-B3(IT,2))))
          DEN=DEN+D1
          T1=T1+(IPC(IT)*D1)
          D2=D2+(((B1(IT,1)-B1(IT,2))*(Y1(IT,1)-Y1(IT,2)))+
     *        ((B2(IT,1)-B2(IT,2))*(Y2(IT,1)-Y2(IT,2)))+
     *        (B3(IT,1)-B3(IT,2))*(Y3(IT,1)-Y3(IT,2))))

          C(IT,1) = B1(IT,2) - B1(IT,1)
          C(IT,2) = B2(IT,2) - B2(IT,1)
          C(IT,3) = B3(IT,2) - B3(IT,1)

  200   CONTINUE
        TD = DSQRT(DEN)
```

```
      DEN=DEN/LEN
      TNOT=(T1/LEN)/DEN
      DELTA=-(D2/LEN)/DEN
```

Calculate m and g.

```
      D1=0.0D0
      D2=0.0D0
      DO 210 IT=1,LEN
         DD=(((B1(IT,1)-B1(IT,2))*(B1(IT,1)-B1(IT,2)))+
     *      (B2(IT,1)-B2(IT,2))*(B2(IT,1)-B2(IT,2)))+
     *      ((B3(IT,1)-B3(IT,2))*(B3(IT,1)-B3(IT,2))))*
     *      (IPC(IT)-TNOT)*(IPC(IT)-TNOT))
         D1 = D1 + DD
         D2=D2+(((B1(IT,1)-B1(IT,2))*(Y1(IT,1)-Y1(IT,2)))+
     *      ((B2(IT,1)-B2(IT,2))*(Y2(IT,1)-Y2(IT,2)))+
     *      (B3(IT,1)-B3(IT,2))*(Y3(IT,1)-Y3(IT,2))))*
     *      (IPC(IT)-TNOT)
  210 CONTINUE
      MEQ=-(D2/LEN)/(D1/LEN)
      TM = D1
      DO 230 J=1,2
         DO 220 IT=1,LEN
            YC1(IT,J)=Y1(IT,J)+(B1(IT,J)*(DELTA+(MEQ*(IPC(IT)-TNOT))))
            YC2(IT,J)=Y2(IT,J)+(B2(IT,J)*(DELTA+(MEQ*(IPC(IT)-TNOT))))
            YC3(IT,J)=Y3(IT,J)+(B3(IT,J)*(DELTA+(MEQ*(IPC(IT)-TNOT))))
            GEQ(IT)=DELTA+(MEQ*(IPC(IT)-TNOT))
  220    CONTINUE
  230 CONTINUE
```

Create the output files.

```
      SSR = 0.0D0
      SSB = 0.0D0
      DO 231 IT = 1,LEN
         SSR = SSR + (YC1(IT,1) - YC1(IT,2))**2 + (YC2(IT,1) -
     *      YC2(IT,2))**2 + (YC3(IT,1) - YC3(IT,2))**2
         D1=(((B1(IT,1)-B1(IT,2))*(B1(IT,1)-B1(IT,2)))+
     *      ((B2(IT,1)-B2(IT,2))*(B2(IT,1)-B2(IT,2)))+
     *      ((B3(IT,1)-B3(IT,2))*(B3(IT,1)-B3(IT,2))))
         SSB = SSB + D1*GEQ(IT)**2
  231 CONTINUE
      MSR = SSR/(LEN - 2)
      MSB = SSB/2
      SIG = DSQRT(MSR)
      TD = TD*DELTA/SIG
      TM = MEQ*DSQRT(TM)/SIG
```

Calculate VDEL,SDEL,VM,SDVM (variances and standard deviations).

```
      DUM2 = 0.0D0
      DUM3 = 0.0D0
      DUM5 = 0.0D0
      DUM6 = 0.0D0
      DO 235 I=1,3
         DO 233 J = 1,3
            DUM1 = 0.0D0
            DUM4 = 0.0D0
            DO 232 IT = 1,LEN
               DUM1 = DUM1 + (C(IT,I)*C(IT,J))
               DUM4 = DUM4+((IPC(IT)-TNOT)**2 *C(IT,I)*C(IT,J))
232         CONTINUE
            DUM2 = DUM2 + (M(I,J) * DUM1)
            DUM5 = DUM5 + (M(I,J) * DUM4)
233      CONTINUE
         DO 234 IT = 1,LEN
            DUM3 = DUM3 + (C(IT,I)**2)
            DUM6 = DUM6 + ((IPC(IT) - TNOT)**2 * C(IT,I)**2)
234      CONTINUE
235   CONTINUE

      VDEL = DUM2/DUM3**2
      SDEL = DSQRT(VDEL)
      VM = DUM5/DUM6**2
      SDVM = DSQRT(VM)

      DUM2 = 0.0D0
      DO 238 IT = 1,LEN
         DUM1 = 0.0D0
         DO 237 I = 1,3
            DO 236 J = 1,3
               DUM1 = DUM1 + (C(IT,I)*C(IT,J)*M(I,J))
236         CONTINUE
237      CONTINUE
         DUM2 = DUM2 + ((IPC(IT) - TNOT) * DUM1)
238   CONTINUE

      COVDM = DUM2/(DUM3 * DUM6)

270   READ(11,'(A)',END=280)LINE
      GOTO 270
280   BACKSPACE 11
      IF(SDEL.EQ.0.0D0) THEN
         DELTAN = SDEL
         MEQN = SDEL
         R = SDEL
         GOTO 285
      ENDIF
      DELTAN = DABS(DELTA)/SDEL
      MEQN = DABS(MEQ)/SDVM
      R = COVDM/(SDEL*SDVM)
285   DELTA = DELTA * 1000.0D0
      MEQ = MEQ * 1000.0D0
      SDEL = SDEL * 1000.0D0
```

```
      SDVM = SDVM * 1000.0D0
      WRITE(11,350)DELTA,SDEL,DELTAN,MEQ,SDVM,MEQN,R,TNOT,LEN

 350  FORMAT(F5.2,2X,F5.4,2X,F6.2,2X,F7.5,2X,F7.6,2X,2(F7.2,2X),
     *        F8.2,2X,I2)

      WRITE(*,*)'      PROGRAM COMPLETED !'
      END



      SUBROUTINE TGEN(LL,G,VV,A2,P1,P2,ANGLE,TIME,
     *                V0,V1,DEPTH,IFLAG)
```

TGEN generates transit time and elevation angle at a target if given the horizontal range, the depth of the sensor and the target, the layer boundaries and the gradients. Isogradiant raytracing is used.

Calling Arguments are as follows:

LL	- An array containing the layer midpoints.
G	- An array containing the gradients for each layer.
V V	- An array containing the velocity at each layer.
A2	- The depth of the sensor (positive down).
P1	- Range of the target (horizontal down).
P2	- Depth of the target (positive down).
V0,V1	- The values for a straight line single layer regression of depth vs. velocity.
DEPTH	- An array containing the depth of each layer.

Return arguments are as follows:

ANGLE	- The final angle at the target.
TIME	- The time of transit.

User Notes:

All floating point numbers are defines : REAL*8

All times are in seconds, and all angles in radians.

```
      DIMENSION L(55),G(55),V(55),LL(55),VV(55)
      DIMENSION TH(55),T(55),VZ(55)
      DIMENSION C2(55),TT(55),DEPTH(55)
      REAL*8 L,G,LL,VV,TH,VZ,C2,TT,T,R0,A1
      REAL*8 A2,P1,P2,C1,C22,THETA,VM
      REAL*8 THETAZ,RV,R,TIME,ANGLE,EP,DZ,DEPTH,V0,V1,V
```

Initialization: Set the value for DZ, the layer thickness. The sensor is assumed to be at RANGE 0. Determine the values for J, which is 1+Number of layers less than or equal to the sensor depth, and I, which is the number of layers less than or equal to the torpedo depth. Redefine the endpoints of those layers locally to be the depths of the torpedo and sensor. Define local values for the LL and VV arrays.

```
      EP=0.1D-6
      DZ=25.0D0
      A1=0.0D0
      J=1
      I=0
      DO 10 K=1,55
        L(K)=LL(K)
        V(K)=VV(K)
        IF(DEPTH(K).LE.A2) J=J+1
        IF(DEPTH(K).LE.P2) I=I+1
 10   CONTINUE
      IF(I.LE.0) I=1
      N=1+J-I
      V(I)=V(I)+G(I)*(P2-L(I))
      V(J)=V(J-1)+G(J-1)*(A2-L(J-1))
      L(I)=P2
      IF(A2.GT.L(J-1)) G(J)=G(J-1)
      L(J)=A2
      R0=P1-A1
      V1=(V(J)-V(I))/(A2-P2)
      V0=V(J) - V1*A2
```

Calculate an initial estimate for the angle and time using a single layer approximation.

```
      C22=-V0/V1
      C1=((0.5D0)*(P1-A1))
      C1=C1+((0.5D0)*(L(I)-L(J))*(L(I)+L(J)-2.0D0*
     *   C22)/(P1-A1))
      IF(IFLAG.GT.0) GOTO 48
      THETA=DATAN((A1-C1)/(A2-C22))
      WRITE(*,*)'    DEFINE THETA AGAIN'
      GOTO 49
 48   THETA=ANGLE
 49   CONTINUE
```

Use the angle THETA to raytrace back through all the layers. First, use the ray invariant (RV) and the velocity to calculate the entrance andgle at each layer..

```
      R=A1
```

Find the maximum sound speed.

```
      VM = V(I)
      DO 495 K = I,J
        IF (V(K).GT.VM) VM = V(K)
 495  CONTINUE

 50   RV=DCOS(THETA)/V(J)
      DO 60 K=I,J
        TH(K)=DACOS(RV*V(K))
```

```
      T(K)=DTAN(TH(K))
      VZ(K)=V(K)-L(K)*G(K)
      C2(K)=-VZ(K)/G(K)
 60   CONTINUE
```

Using the angle just calculated, iterate backwords through the layers from sensor to target to get the horizontal range. Stop at the depth of the target.

```
      R=0.0D0
      DO 70 K=J,I+1,-1
         C1=R-T(K)*(L(K)-C2(K-1))
         R=C1+T(K-1)*(L(K-1)-C2(K-1))
 70   CONTINUE
```

Test if the value for the range is within torerance. If not, redefine THETA, the initial angle, and raytrace again. If wthin tolerance, calculate the time of travel based on THETA, and return.

```
      EP=0.1D-6
      IF((DABS(R-P1)).LE.EP) GOTO 100
      THETAZ=THETA
      THETA=DATAN(DTAN(THETAZ)*(R-A1)/R0)
      GOTO 50
100   TT(J)=DLOG((1.0D0+DSIN(TH(J)))/(DCOS(TH(J))))
      TIME=0.0D0
      DO 110 K=J-1,I,-1
         TT(K)=DLOG((1.0D0+DSIN(TH(K)))/(DCOS(TH(K))))
         TIME=TIME+(TT(K)-TT(K+1))/G(K)
110   CONTINUE
      ANGLE=THETA
      RETURN
      END
```

REFERENCES

[1] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J. V. Sanders, *Fundamentals of Acoustics*. New York: Wiley, 1982.

[2] R. R. Read, *Program for the Simultaneous Estimation of Displacement and Orientation Corrections for Several Short Base Line Arrays*, NPS Technical Report, NPS55-85-028, November, 1985.

[3] R. R. Read, *A Study of Underwater Sound Ray Tracing Methodology*, NPS Technical Report (in preparation).

[4] H. D. Patterson and R. Thompson, "Recovery of inter-block information when block sizes are unequal." *Biometrics*, Vol. 58, No. 3, 1971.

INITIAL DISTRIBUTION LIST

Library (Code 0142) Naval Postgraduate School Monterey, CA 93943	2
Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
Department of Operations Research Naval Postgraduate School Monterey, CA 93943 ATTN: Prof. Robert R. Read, Code OR-Re	20
Department of Operations Research Naval Postgraduate School Monterey, CA 93943 ATTN: Prof. R. N. Forrest, Code OR-Fo	1
Naval Postgraduate School Monterey, CA 93943 ATTN: Research Administration, Code 012	1
Department of Operations Research Naval Postgraduate School Monterey, CA 93943 ATTN: Prof. P. Purdue, Code OR-Pd	1
Department of Operations Research Naval Postgraduate School Monterey, CA 93943 ATTN: Prof. J. Esary, Code OR-Ey	1
Department of Operations Research Naval Postgraduate School Monterey, CA 93943 ATTN: Prof. O. B. Wilson, Code PH-Wl	1

Department of Operations Research 1
Naval Postgraduate School
Monterey, CA 93943
ATTN: Prof. L. Johnson, Code OR-Jo

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: R. Mash, Code 50

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: R. D. Helander, Code 51

Naval Undersea Warfare Engineering Station 2
Keyport, Washington 98345
ATTN: S. L. McKeel, Code 512

Naval Undersea Warfare Engineering Station 2
Keyport, Washington 98345
ATTN: J. Knudsen, Code 5122

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: T. Ward, Code 5122

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: G. Olsen, Code 5122

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: J. Hall, Code 5121

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: R. Evans, Code 5113

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: L. Beer, Code 5113

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: L. Bogan, Code 5111

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: D. Hutchins, Code 5232

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: J. Cain, Code 70E2

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: E. Roberts, Code 70B1

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: J. Chase, Code 70B1

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: Cmdr Dewry, Code 80

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: A. Pickard, Code 801

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: P. Correa, Code VITRO

Naval Undersea Warfare Engineering Station 1
Keyport, Washington 98345
ATTN: B. Boyer, Code VITRO

Center for Defense Analyses 1
1800 N. Beauregard St.
Alexandria, VA 22311

Center for Naval Analyses 1
4401 Ford Ave.
Alexandria, VA 22302-0268