AFGL-TR-88-0032
ENVIRONMENTAL RESEARCH PAPERS, NO. 994

Short Term Forecasting of Cloud and Precipitation

A. BOHNE
F.I. HARRIS
P.A. SADOSKI
D. EGERTON

15 January 1988

ATMOSPHERIC SCIENCES DIVISION PROJECT 6670
AIR FORCE GEOPHYSICS LABORATORY
HANSCOM AFB, MA 01731

"This technical report has been reviewed and is approved for publication"

FOR THE COMMANDER

Kenneth M. Glover

KENNETH M. GLOVER, Chief
Ground Based Remote Sensing Branch
Atmospheric Sciences Division

Robert A. McClatchey

ROBERT A. MCCLATCHEY, Director
Atmospheric Sciences Division

This document has been reviewed by the ESD Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

Qualified requestors may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify AFGL/DAA, Hanscom AFB, MA 01731. This will assist us in maintaining a current mailing list.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFGL-TR-88-0032 ERP, No. 994	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Air Force Geophysics Laboratory 6b. OFFICE SYMBOL (If applicable) LYR	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Hanscom AFB Massachusetts 01731-5000	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. 62101F; PROJECT NO. 6670; TASK NO. 10; WORK UNIT ACCESSION NO. 21

11. TITLE (Include Security Classification)
Short Term Forecasting of Cloud and Precipitation

12. PERSONAL AUTHOR(S)
Bohne, A., Harris F.I.*, Sadoski, P.A., Egerton, D.*

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Sep 86 TO Sep87	1988 January 15	104

16. SUPPLEMENTARY NOTATION
*ST Systems Corporation, Lexington, Massachusetts

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Nowcasting, Forecasting, Contour, Segmentation, Pattern Recognition, Radar, Extrapolation, Data Filtering, (SRP)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A methodology for real-time operations has been developed for the short-term forecasting of cloud and precipitation fields. Pattern recognition techniques are employed to extract useful features from the data field and extrapolation techniques are used to project these features into the future. To reduce computational load, contours defined by directional codes are used to delineate features. These contours are subdivided and attributes such as length, location, and location of each segment are determined. Segment matching is performed for successive observations and attribute changes are monitored over time. Several techniques for the forecasting of attributes have been explored, and an exponential smoothing filter and a linear trend adaptive smoothing filter have been chosen as most appropriate. Currently analysis is performed on a minicomputer and image processor system utilizing radar reflectivity data. Refinement of these techniques and extension into a more comprehensive short term forecasting program is planned.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
Alan R. Bohne	617-377-2943	LYP

DD FORM 1473, JUN 86 *Previous editions are obsolete.*

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Contents

Contents

Illustrations

Tables

Tables

Short Term Forecasting of Cloud and Precipitation

1. INTRODUCTION

The Air Force Geophysics Laboratory is developing techniques for accurate nowcasting of cloud and precipitation. Such a capability would be useful in supporting general air and terminal operations and in predicting periods of communication signal loss. The communication aspect is particularly important for satellite to ground links utilizing very high frequencies since significant degradation of signal can result due to intervening cloud or precipitation. The current effort represents the development of a computer-based methodology for using continuously updated radar and satellite data as input for 0 - 0.5 hour forecasts of future cloud and precipitation fields. The underlying premise of the procedure is that future fields may be derived from forecasting selected field intensity contours. This requires monitoring the evolving patterns of field intensity and, with knowledge of the spatial changes of these contours over time, extrapolating to future cloud and precipitation fields.

The program developed along two lines. One primary effort was the acquisition and integration of hardware for data assimilation and analysis and the development of a suitable system software environment. The second effort was the development of the analysis techniques and associated software. The discussion of hardware and associated system software is presented in a separate report.[1] (Sadoski et al, 1987). This report describes the data analysis procedures that have been developed. A discussion of potential techniques for data analysis and forecasting was presented

---

(Received for Publication 13 January 1988)

1. Sadoski, P.A., Egerton, D., Harris, F.I, and Bohne, A.R. (1988) *The Remote Atmospheric Probing Information Display (RAPID)*, AFGL-TR-88-0036, AD A196314.

earlier by Bohne and Harris[2], where a candidate processing methodology was presented. The methodology was essentially broken ' .to four steps: (1) data preprocessing, (2) feature definition and extraction, (3) attribute definition and mapping, and (4), attribute forecasting and feature reconstruction. In this report a discussion of each of these steps will be presented.

2. DATA PREPROCESSING

2.1 Data Preprocessing Considerations

Prior to the application of feature extraction, mapping, and forecasting routines, much consideration must be given to providing a data set that supports these efforts. During development and testing of potential software techniques, it was quickly realized that successful performance of the various methods was correlated with the quality of the input data. Various steps must be employed to develop a well-behaved, yet representative, data set on a suitable coordinate system; specifically, filtering procedures to interpolate data, smooth boundaries, remove errors, and fill gaps must be employed.

Selection of routines and their specific manner of application are somewhat dependent upon the hardware environment. The hardware environment is defined by the Remote Atmospheric Processing and Interactive Display (RAPID) System. The RAPID System has as its main components Digital Equipment Corporation VAX minicomputers, an ADAGE 3000 image processor, and associated peripherals and communication links. For details regarding the system hardware and software and the resulting environment see Sadoski et al.[1] RAPID ingests radar and satellite data from other computer systems. These data are then processed and displayed for the forecast problem within RAPID.

2.2 Data Coordinate Transformations

The raw data received from satellites and radars are in very different coordinate systems. The radar collects its data in a three-dimensional spherical framework, along radials emanating from the radar itself. The data from the GOES satellite are organized in a distorted planar framework, the distortion being due to the oblique viewing angle of the satellite sensors relative to the curving earth's surface and the mapping of features onto the satellite's flat viewing plane. It is essential that the data from these two systems be converted to a common grid if they are to be used together in a quantitative forecast system. The grid center was chosen to be collocated with the radar at the Ground Based Remote Sensing Branch (LYR) of AFGL. Because the forecast area is limited to a range of about 250 km about the grid center and a regular grid system is most appropriate for data manipulation and display by the image processor, the grid was selected to be rectangular Cartesian.

---

2. Bohne, A.R. and Harris, F. Ian (1985) *Short Term Forecasting of Cloud and Precipitation*, AFGL-TR-85-0343, AD A169744.

2.2.1 RADAR DATA

Interpolation of radar data between coordinate systems can be a very computer intensive operation in terms of time and memory usage. However, Mohr and Vaughan[3] devised a very efficient algorithm that minimizes these requirements. Basically, their interpolation algorithm can be divided into three task areas: data preprocessing, ingestion, and interpolation. In the preprocessing phase the user defines the rectangular Cartesian coordinate system onto which interpolated values will be placed. For each Cartesian grid point, the spherical coordinates (range, azimuth, and elevation) are computed. These are then sorted and subsequently stored on disk in the form of level files. Each level file contains the spherical and Cartesian coordinates for all Cartesian grid points between two consecutive elevation scans of the radar. Thus, as data from a particular elevation scan are read from the radar processor during the ingestion phase, only two level files need to be searched to determine data placement on the Cartesian grid. Also, because the spherical coordinates are known for each Cartesian grid point, it is unnecessary to compute the Cartesian coordinates for each radar data point as is the case for most conventional interpolation techniques. Thus, a substantial reduction in the number and complexity of the calculations is obtained. Interpolations to the Cartesian grid can be made using either a bilinear or nearest grid point technique, both of which lend themselves very well to this coordinate sorting method. Once calculations are completed in all level files the resultant interpolated data are sorted into the horizontal planes of the rectangular framework and then output to disk or ADAGE display memory.

The National Center for Atmospheric Research developed a software package in which one of the elements is an interpolation routine that utilizes the Mohr and Vaughan algorithm. While this software package was obtained by AFGL and used to produce data for other aspects of this study, the interpolation routine within this package is both cumbersome and slow. This results from the software being designed for versatility and not for speed.

As a consequence, an approach that is more efficient and tailored to the AFGL hardware configuration is required for real-time implementation of the Mohr and Vaughan algorithm. It was determined that the interpolation would be performed on the Perkin-Elmer (PE) 3242 which is directly linked to the Remote Sensing Branch radar processor. The resultant Cartesian fields are then transferred to RAPID. This approach was taken because the PE 3242 has more memory, is computationally faster, and has more disk storage capability than RAPID. The VAX is therefore freed from this time consuming operation, allowing for more efficient management of data processing by RAPID.

2.2.2 SATELLITE DATA

The intent of the program was to utilize satellite imagery data in those regions not effectively interrogated by radar. This occurs primarily in storm top regions near the radar, and in the effectively clear-air boundary layer. Data from the GOES satellite can be shipped over a DECnet link from the Satellite Branch [illegible] at AFGL to the VAX minicomputers. The satellite imagery data,

---

3. Mohr, C.G. and Vaughan, R. (1979) An economical procedure for Cartesian interpolation and display of reflectivity factor data in three-dimensional space, *J. Appl. Meteorol.* **18**:661-670.

obtained once every half hour, are first preprocessed at LYS to reduce the field of view to roughly a 500 by 500 km region centered over the LYR radar. The visible and infrared imagery data have resolutions of about 1 km and 4 km, respectively. The data are acquired in the satellite viewing plane and, thus, include mapping effects from the spherical surface of the earth. Software was developed to transform the satellite data from this distorted reference frame to the same Cartesian reference frame in which the radar data are analyzed. Because of inaccuracies in determining precise satellite position and orientation, the data are also registered, that is, further translated and warped to fit known map locations on the Earth's surface. Once the satellite fields have been transformed to the Cartesian reference grid system, the data can then be output to disk or moved into ADAGE display memory. However, because of the very coarse temporal resolution, these data are used primarily to develop a general overview of the meteorological situation and are not explicitly incorporated into the extrapolative forecasting process.

2.3 Evolution and Vertical Advection Considerations

Of concern in the nowcasting process is the minimum spatial scale that can effectively support the forecasting process. This requires consideration of the temporal and spatial scales of both the observed meteorological phenomena and of the collected data. Certainly, with slowly evolving stratiform systems, little difficulty is expected in forecasting field motion and evolution. However, difficulty may be expected with convective phenomena where significant convective cells can have spatial extents of only 2 to 5 km and lifetimes of 6 to 30 min.[4] With the temporal resolution of the radar data no better than 5 min and the spatial resolution as coarse as 3 km or more at long range, it is an unreasonable expectation that all individual convective elements may be effectively tracked. This was demonstrated by Harris and Petrocchi[5] who concluded that automated tracking of these features at single elevation levels was unreliable due to the evolution and vertical motions of the precipitation distributions.

As an illustration of these effects, but on a larger scale than addressed by Harris and Petrocchi, the 32 dBZ radar reflectivity factor contours at 5 km above sea level are plotted in Figures 1a-d for four successive scans obtained during passage of Hurricane Gloria through New England on 27 Sept 1985. These scans are each 6 minutes apart. The original data were collected every 300 m in range, 0.6 degree in azimuth, and 0.8 to 2.2 degrees in elevation and interpolated to a Cartesian grid with 2 km horizontal resolution using bilinear interpolation. The degree of evolution of the contours between observations suggests that use of contours at a single level will not support forecasts of any reasonable detail. One approach to mitigate this problem would be to obtain some assessment of vertical advection (but probably not of precipitation growth) by monitoring the correlation of changes at successive height levels. This process is very complex and computationally costly. Another approach

---

4. Foote, G.B. and Mohr, C.G. (1979) Results of a randomized hail suppression experiment in northeast Colorado. Part VI: Post hoc stratification by storm intensity and type, *J. Appl. Meteorol.* **18**:1589-1600.

5. Harris, F.I. and Petrocchi, P.J. (1984) *Automated Cell Detection as a Mesocyclone Precursor Tool*, AFGL-TR-84-0266, AD A154952, 32 pp.

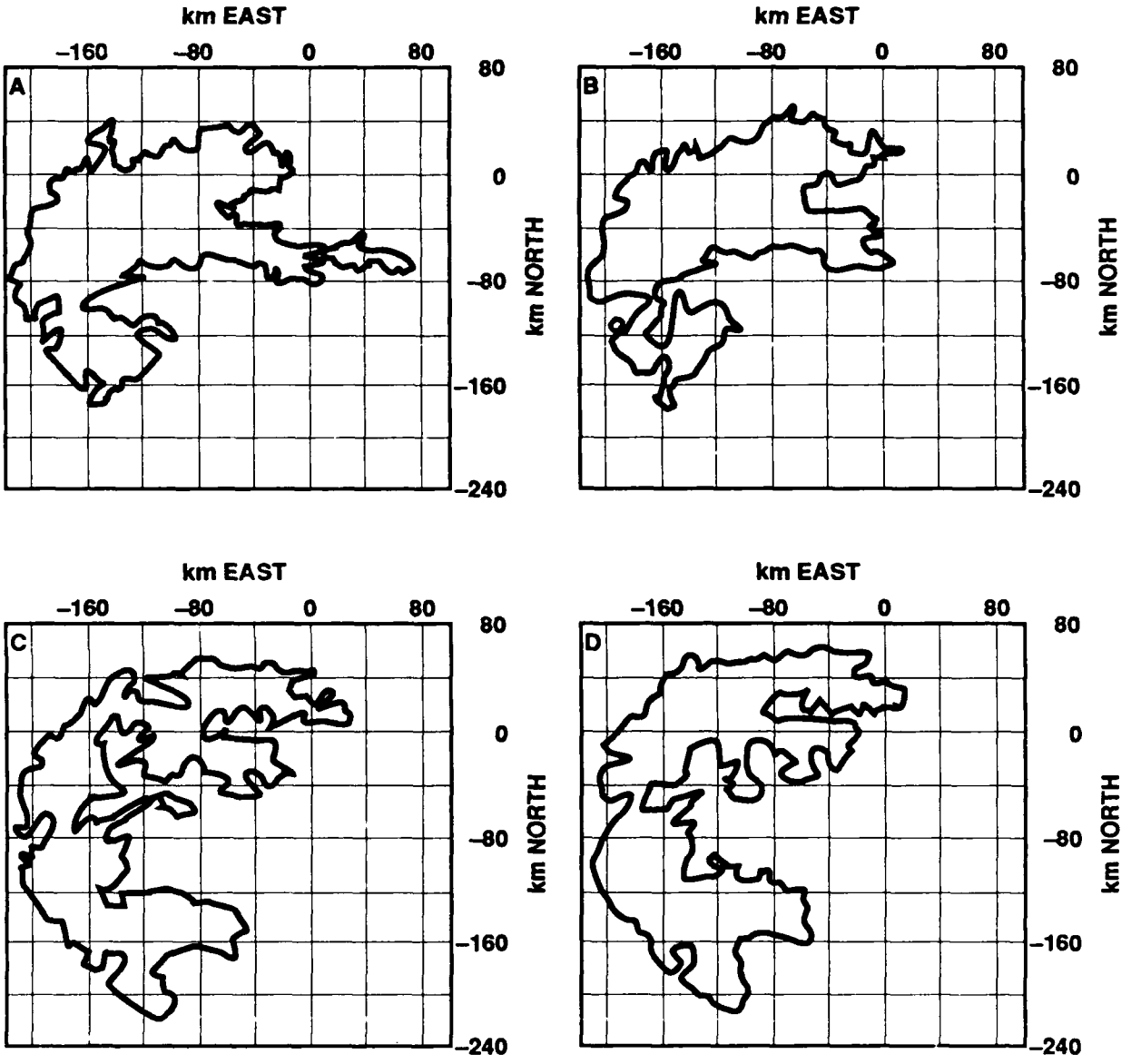

Figure 1. The 32 dBZ reflectivity contour at 5.0 km height at (a) 1052 EST, (b) 1058 EST, (c) 1114 EST, and (d) 1121 EST

is to employ a more conservative field, possibly obtained through limited vertical integration or compositing of the radar reflectivity data.

The desire to ultimately produce forecasts in real time has led to use of the compositing approach in RAPID. An example of the effects of compositing the radar data are shown in Figures 2a-d where the grid values were determined by retaining the maximum reflectivity factor throughout the depth of observations above a horizontal grid point. While there is still significant fine-scale structure, the field evolution is certainly less explosive, easier to follow in time, and more meaningful in terms of the forecasting process. Although a vertical averaging method (that is, averaging data above and below the grid point) is not currently employed, this method can also produce a fairly conservative field. However, the grid values are biased towards smaller reflectivity values and may underestimate the true hazard potential.

2.4 Data Editing

Once the data have been interpolated and composited, further preprocessing is still required to remove noise, smooth boundaries, de-emphasize small-scale features, and fill any existing data gaps. These processes generally involve passing filters of varying types and shapes across the data field. Several that have been implemented will now be briefly discussed.

2.4.1 MEDIAN FILTERING

The first step in the processes of noise suppression and data smoothing is the application of a median filter. The filter is of some specified geometric shape (window) with an odd number of elements. The center data value within the window area is replaced by the median of all values in the window. The filter can be passed across the entire data field, sometimes as many as 2-3 times in succession, to achieve the desired effect. Some more commonly used windows for a median filter are shown in Figure 3. The RAPID methodology employs the 3×3 box to remove noise spikes often seen in radar data and to smooth field intensity contours. The 1×5 line filter is used to eliminate missing scan lines often found in satellite imagery data.

Median filtering is highly effective for noise suppression. In general, regions that are unchanged after a single pass of the median filter will remain unchanged in subsequent passes.[6] With image data quantized into various discrete levels, only contour edges will be affected by successive median filtering, since these areas are the only locations exhibiting field change. The effect of a simple 3×3 filter is demonstrated in Figure 4.

A two-dimensional filter such as the 3×3 box filter generally preserves edges but removes thin lines, raggedness, and smoothes out corners. A plus-shaped filter generally preserves horizontal and vertical lines and corners but removes diagonally oriented lines and corners. Thus, small-scale (relative to the size of the filter) perturbations of a feature will be smoothed by a median filter.

Figures 5a-d show the results of applying a 3×3 median filter to the composited data used to generate Figures 2a-d. The filter eliminates the small scale raggedness along the contour, resulting in

---

6. Pratt, William K. (1978) *Digital Image Processing*, Wiley-Interscience, New York.

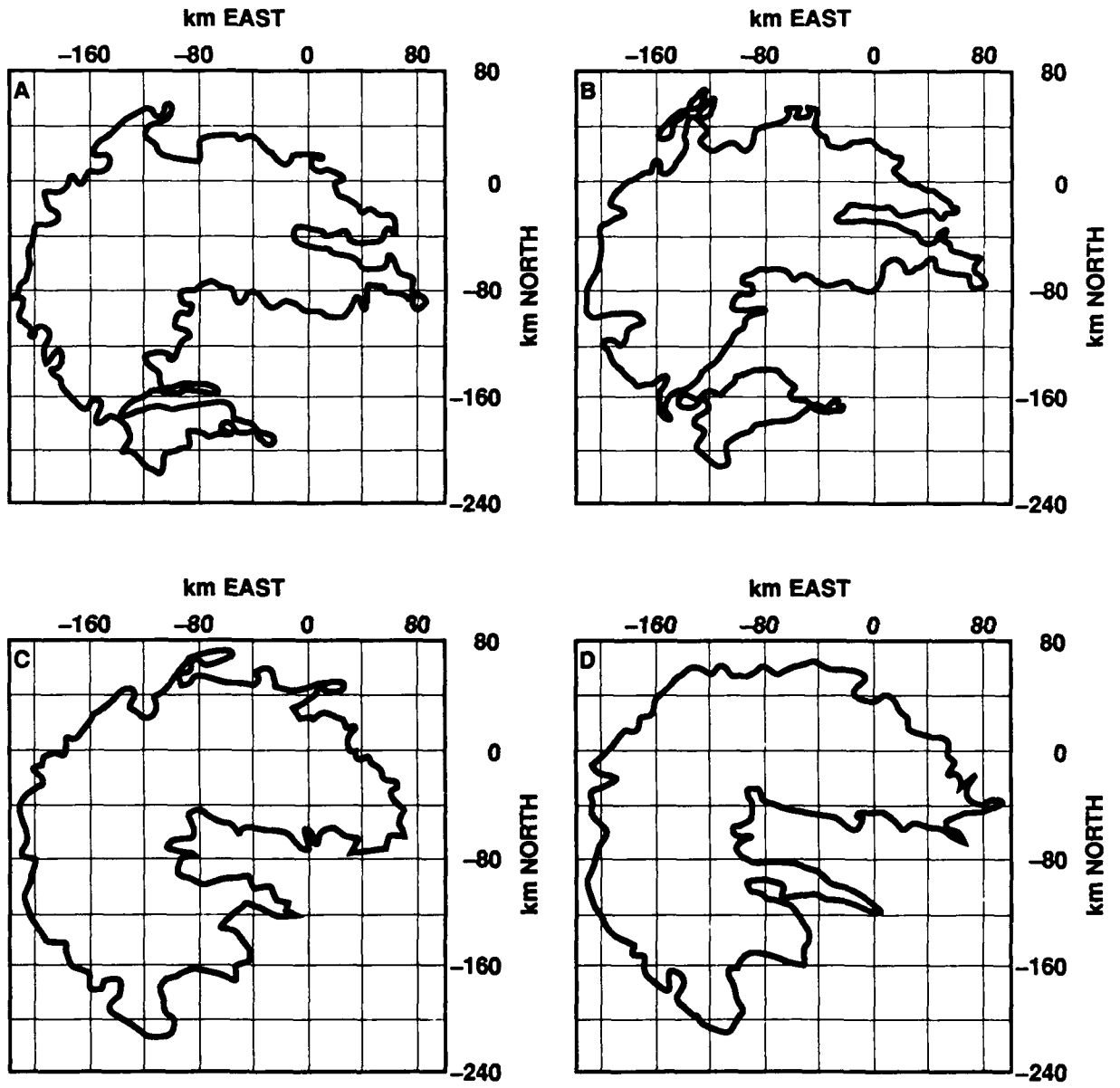

Figure 2. The 32 dBZ reflectivity contour for composited reflectivity - times as Figure 1

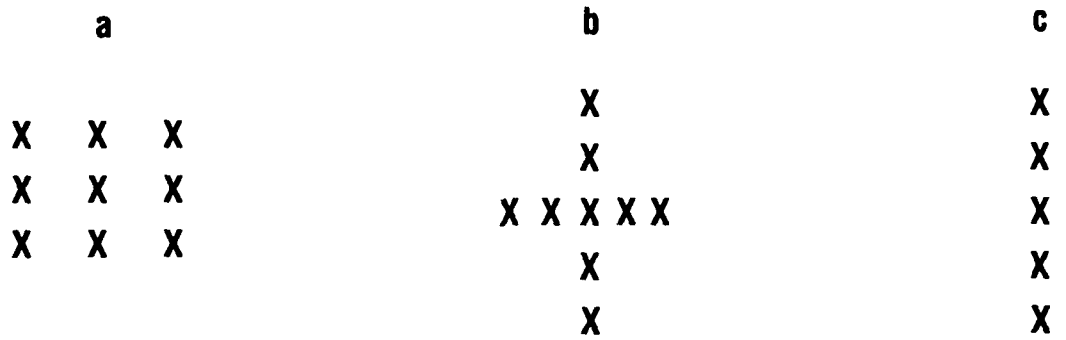

Figure 3. Windows commonly used with median filters: (a) standard areal filter, (b) gap filter and, (c) horizontal line filter

FILTER =

X	X	X
X	X	X
X	X	X

ORIGINAL FEATURE					1PASS WITH FILTER					2 PASSES WITH FILTER				
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	0	0	0	1	0	1	0	0	1
1	1	1	0	1	1	1	0	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 4. Example of effect of a 3×3 median filter on a binary image

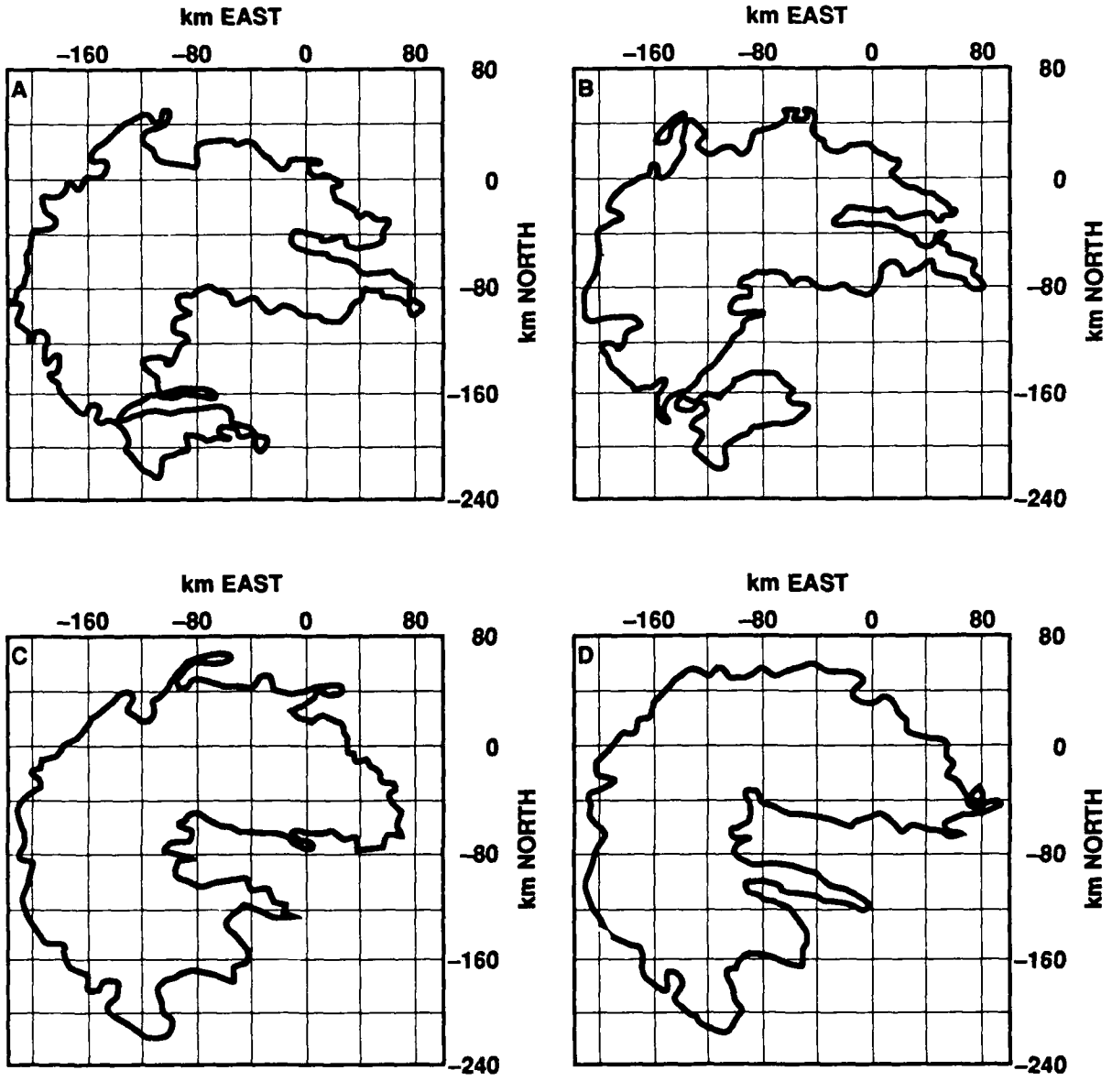

Figure 5. Same data as in Figure 2 but with a 3×3 median filter applied before contouring

a more conservative field. However, the contour structure in this particular example is still considered more complex than can be effectively handled in current tracking and forecasting schemes. This is an example where further applications of the filter are required.

One problem that arises with use of a median filter is that its response function is basically unknown. This is because the data value selected to replace the center value can be from any location within the window area. Thus, there is quantitative uncertainty as to its effects upon the distribution of small-scale features within the data. However, since the goal is to achieve a conservative field contour and the filter does preserve the larger scale shape of the contours while smoothing the smaller scale noise, the effects are considered, and have so far been observed, to be unimportant.

2.4.2 LOWPASS FILTERING

The lowpass filter is also used for noise suppression by preserving large spatial scales while suppressing those of relatively short length. The key difference between the lowpass and median filters lies in the fact that while the median filter performs a simple replacement where necessary, the lowpass filter is an averaging filter. It always replaces the center value within the filtering window with some weighted average of its neighbors. Thus, the lowpass filter smoothes the entire field of data. Because the filter weights within the window are precisely known, the filtering effects on the spatial distribution of the data can be determined. Some typical filters are presented in Figure 6 (top row) with the the form which has been adopted in RAPID shown in Figure 6 (bottom row).

There are several reasons why this particular filter was selected. Experience showed a larger (more than 3×3) filter is required for effective noise removal, but the filter size also must be kept as small as possible for computational efficiency. Also, the filtering is to be performed in the image processor, which cannot easily handle division. Therefore, operations are greatly facilitated if a power of two is used as the divisor, for this reduces the division to a shift-left operation. For example, an integer divide by 16 is equivalent to a shift-left by 4 bits. This filter is generally applied only once to the data to minimize the potential for overfiltering. This lowpass filter was applied to the composited field shown in Figures 5a-d with the resultant fields shown in Figures 7a-d. The resultant contours are much smoother but an appendage originally located in the lower middle segment of Figure 5b has disappeared. Filtering has resulted in the elimination of the narrow neck portion of this feature. If the analysis had been extended to define contours for all features within the data set, the more substantial portion of this appendage should appear as a separate feature.

This filter has the potential for propagating contour positions to neighboring grid points, causing a contour to expand or shrink in size. This is particularly true when the field gradients are large and highly nonlinear. Thus, this filter may not preserve the character of a contour as well as the median filter. Also it is computationally more burdensome than the median filter. However, for data fields where contour continuity is not readily apparent, for example when field gradients are small and the noise contribution is large, this filter is very useful for extracting the contours from the noise.

2.4.3 FEATURE EDITING

Occasionally there will be small features present in the data field that should not be tracked. Some examples would be radar ground clutter return or very small isolated precipitation elements that may fade in and out between observations. To eliminate these features contour boundaries are

$$\frac{1}{9} \star \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix} \qquad \frac{1}{16} \star \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix} \qquad \frac{1}{48} \star \begin{vmatrix} 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & 4 & 2 & 1 \\ 2 & 4 & 8 & 4 & 2 \\ 1 & 2 & 4 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{vmatrix}$$

$$\frac{1}{16} \star \begin{vmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & 0 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{vmatrix}$$

Figure 6. Commonly used lowpass filters in upper row and RAPID formulation in lower row

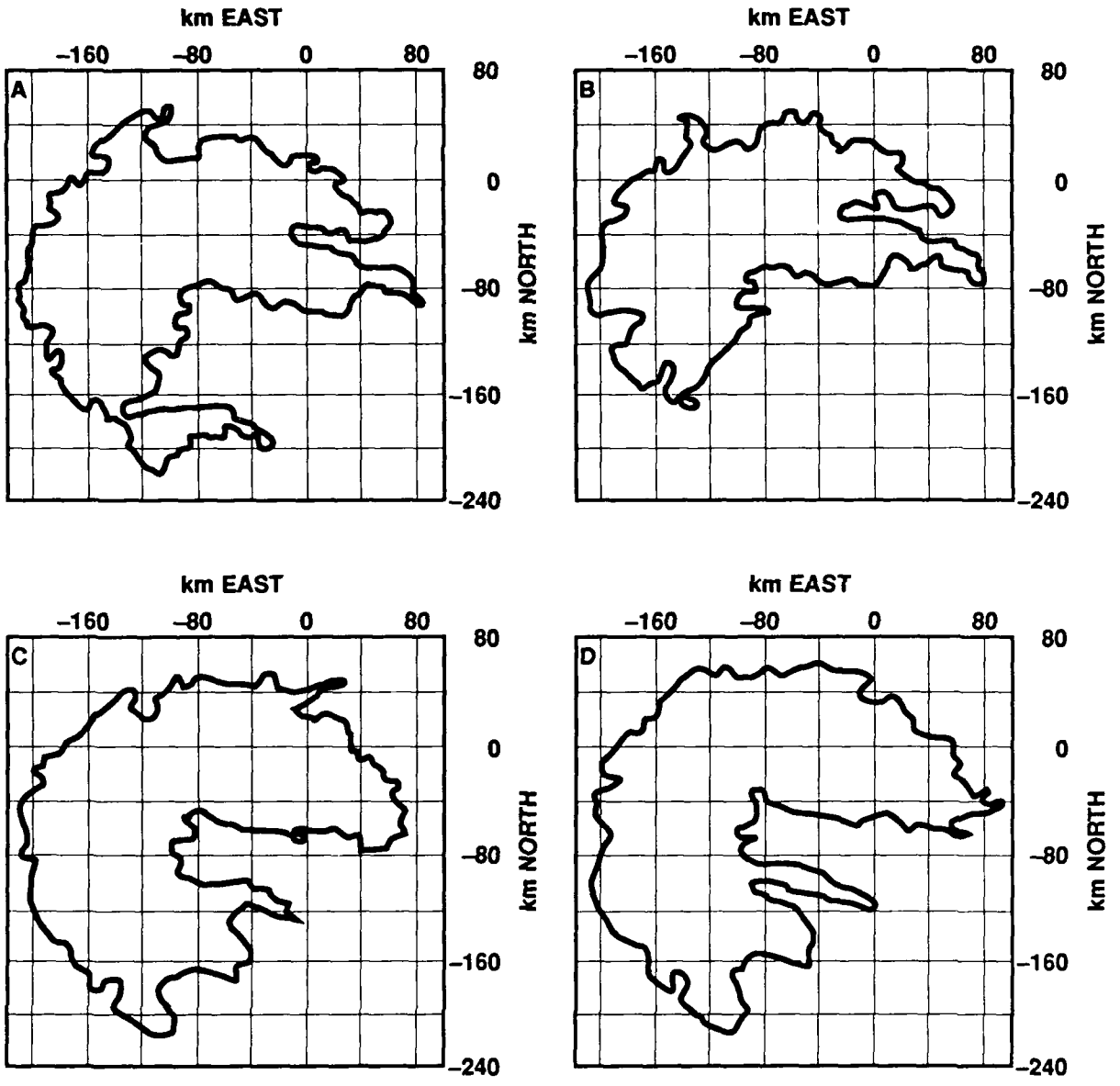

Figure 7. Same data as in Figure 2 but with both a 3×3 median and lowpass filter applied before contouring

interrogated to determine their length (in pixels). If the boundary size of the region is smaller than some predetermined value, then the region is considered to be a contaminant and is eliminated by replacing it with the neighboring field mean value (or lower threshold value if the data are quantized into discrete threshold levels).

2.4.4 GAP FILLING

Missing data may be accounted for in a number of ways, the simplest being replacement with a value determined from the neighboring data field. The gaps usually reside in low radar reflectivity factor regions, typically along the storm boundaries or internal regions where the missing data are essentially surrounded by useful data. Care is taken to not extend storm boundaries unnecessarily, or to fill in regions where the radar did not scan completely (for example, near storm top). Typically, gap filling requires minimal intervention and usually is performed during standard median or lowpass filtering of the data during noise removal.

3. FEATURE MAPPING AND ATTRIBUTE DEFINITION

3.1 Feature Forecasting Considerations

The techniques developed here are generic in the sense that they may be applied to any two-dimensional field of data with distributed values. The methods have been tested on data from both radar and satellite sensors. The initial intent was to merge satellite and radar data into a single data field that would be useful for forecast generation. However, radar measures reflectivity factor from the precipitation throughout the storm volume while a satellite measures visible and infrared radiance from the cloud boundaries. Since there is no good physical relationship between these measurements, numerical combination(s) of the three fields is really not possible. Qualitative merging of the fields is possible if the data are composited in a binary fashion (that is, if grid value is above a threshold value, the grid value is set to 1) or contours of the respective fields are overlaid. However, the magnitudes and spatial distributions of the individual fields are lost. Also, the temporal resolution of the radar and satellite data are 5 - 7 min and 30 min, respectively. Neither overlaying an evolving radar data field on a stationary satellite field nor discarding radar data not coincident with satellite data can be considered as reasonable approaches. Since many of the expected applications are affected primarily by precipitation rather than by cloud, and because satellite data are of little use for the forecasting of rapidly evolving precipitation fields, primary emphasis was placed upon use of radar data.

There are two basic philosophies that may be adopted to derive forecasts of precipitation field motion and evolution from the radar data. One is to perform correlation analyses over limited areas, until the entire data set has been interrogated.[7,8] These techniques work extremely well for tracking

---

7. Rinehart, R.E. (1979) *Internal Storm Motions from a Single Non-Doppler Weather Radar*, NCAR/TR-146+STR, National Center for Atmospheric Research, Boulder, CO, 262 pp.
8. Smythe, G.R. and Zrnic', D.S. (1983) Correlation analysis of Doppler radar data and retrieval of the horizontal wind, *J. Clim. Appl. Meteor.* **22**:297-311.

precipitation features having scales of the order of 5 km. However, they are notoriously slow,[9] even with limited data sets. Because the 0 to 30 min forecast problem requires utilization of every available radar scan sequence, and each sequence may result in as many as three two-dimensional fields (three selected altitudes) of data, with each field containing up to 256×256 grid points, correlation techniques would not support real-time operations and are not employed. However, this approach may be reconsidered for a 0 to 2 hr forecast effort to be addressed in a future task.

The second philosophy is to extract features from the data, determine their attributes, and forecast changes of these attributes. These attributes might include such quantities as centroid positions or storm boundaries. The intent here is to reduce the amount of data tracked and forecast, while still retaining an accurate description of the spatial distribution of the precipitation field. The normal mode of operation for an observer is to view the field of data and first locate features such as maxima, minima, and gradient zones. That is, the observer mentally thresholds the data to determine the spatial structure. This natural method of using thresholds to delineate areas of selected intensity, with each such area identified as a feature, is employed in RAPID.

3.2 Data Representation Considerations

There are several ways one might represent the features and derive characteristics that could then be used for generating predictions. One approach is to utilize binary data values where grid points with values below the threshold are given a value of 0 and those above a value of 1. The result is a region(s) of 1s delineating the desired feature(s) surrounded by a field of 0s. While this method certainly makes features easily identifiable, the resulting data set may still be large and require significant computational time and power.

Another approach is to simply follow the contour of the feature, as defined by some threshold data value, and retain only sufficient information to describe the contour. This approach then reduces the two-dimensional intensity distribution to a set of user-selected contour lines. The working data set size is significantly reduced with the degree of reduction being dependent upon the method used to describe the contour. Judicious selection of contour thresholds allows for adequate description of the data field. Because of the significant restrictions on computer memory, and analysis and forecast update time, this approach was chosen.

3.3 Contour Extraction

Three techniques were considered for extraction and description of the feature contours: specifically, use of interpolated Cartesian contours, geometric approximation methods, and the Freeman chain code. These will be discussed in terms of their formulation and advantages.

---

9. Smythe, G.R. and Harris, F.I. (1984) *Sub-Cloud Layer Motions from Radar Data Using Correlation Techniques*, AFGL-TR-84-0272, AD A156477.

3.3.1 INTERPOLATED CARTESIAN CONTOUR AND GEOMETRIC APPROXIMATION

There are numerous methods that delineate contours by employing interpolation procedures to precisely locate the coordinates of contour points. Generally the data are searched until a value is found that exceeds the desired contour value, and the contour locations about this point are then determined by interpolating between the current position value and data on neighboring grid points. The interpolation filter can have a variety of forms such as uniform (average), exponential, and Cressman. The number of points employed in the filter can be as few as four or as many as contained in the entire field. Obviously, the more complicated the filter and the more points used, the greater will be the amount of time and memory required for data processing. This technique generally results in the most accurate determination of the location of a contour, for it allows for interpolation between grid points. The resulting data set, a two-dimensional array of the contour coordinates,[10] are stored for further processing. This technique is somewhat cumbersome and the interpolation procedure can be time consuming if the contour is large and complex. Thus, it is not considered viable in the rapid update time frame considered here.

The second technique, geometric approximation, involves fitting a variety of sizes of specific geometric figures such as ellipses to the feature. This technique has been used in forecasting synoptic scale pressure patterns[11] and heights of pressure surfaces.[12] The technique is especially suited to these types of feature forecasts since representations of low and high pressure systems tend to be somewhat elliptical with smoothly varying contours. Basically, the technique involves locating the centers of all maxima (and minima for synoptic data) within the data set. Then with an adaptive nonlinear, least-squares algorithm,[13] ellipses are fitted to the desired contour(s), one for each maximum. The family of fitted contours are the equivalent representation of the original contour set. Five parameters of the ellipse are allowed to vary: the coordinates (2) of the center of the feature, the length of the major axes, eccentricity, and orientation angle of the ellipse.

Generally, this technique has been used for features that are relatively conservative in nature, that is, those that evolve slowly between observations. Also the contours generally have simple overall shapes and vary smoothly. Unfortunately, as often observed from radar echoes there is usually a high degree of evolution in the precipitation patterns within storms and the contour shapes are often complex. To make this method robust enough to accommodate these types of data would require excessive field smoothing to force the contours to adhere to the constraints of simplicity and slow change. Thus, although this method is attractive by allowing for significant data reduction, the equivalent representation can be in error and thus this technique is not employed.

---

10. Dudani, S.A. (1976) Region extraction using boundary following, *Pattern Recognition and Artificial Intelligence* (C.H. Chen, editor), Academic Press, Inc., New York, NY, pp. 216-232.
11. Clodman, S. (1984) Application of automatic pattern methods in very-short-range forecasting, *Proc. Nowcasting II Symposium*, Norrkoping, Sweden, ESA SP-208.
12. Williamson, D.L. and Temperton, C. (1981) Normal mode initialization for a multilevel grid-point model, Part II: Nonlinear Aspects, *Mon. Wea. Rev.* **109**:744.
13. Dennis, J.E., Gay, D.M., and Welsch, R.E. (1977) *An Adaptive Nonlinear Least-Squares Algorithm*, Cornell Computer Science TR77-321.

3.3.2 FREEMAN CHAIN CODE

A simpler concept was introduced by Freeman.[14] As in the interpolated Cartesian contour method, a threshold is applied to the data. However, for the Freeman technique one needs only to determine whether grid point data values are above or below the threshold value. This effectively reduces the data to binary form and makes the desired feature distinct from the background. The contour of the feature is extracted and stored as a one-dimensional directional array, with each array entry having only one of eight possible values.

To extract the boundary and form the directional array the two-dimensional data field is scanned in a left-to-right, top-to-bottom manner until a point on the boundary of the contour is located. This is referred to as the starting point, or origin, of the contour. Its (x,y) position in the grid is saved as the first element in the contour array. From this grid point the contour of the region is determined by keeping the feature within the contour always to one side of the path being followed (for example, to the left) while searching for the next nearest grid point on the contour. Once the next point is found its location may be represented by the directional code presented in Figure 8. In this figure, the x represents the original contour point and the numbers represent the direction code (angle) from that point to the next contour point. Because the grid is regular, these eight directions (angles) represent all the possibilities that one might encounter. Thus, one simply walks around the feature, determining a directional code for each boundary point until the starting point is again encountered.

The codes leading to the nearest neighbors for all newly located boundary points are saved in an array, along with the origin and final length of the code. Collectively, these three elements completely describe the contour of any closed region and the two-dimensional data array has been reduced to the one-dimensional representation of the Freeman chain code.[15]

An example of the process is shown in Figure 9. The boundary of the rectangle is completely described by:

1) the origin is (1,1)

2) the number of elements in the directional code array is 18

3) the directional code array is { 3, 3, 3, 5, 5, 5, 5, 5, 5, 7, 7, 7, 1, 1, 1, 1, 1, 1 }.

Thus, a 28 point two-dimensional array (56 elements) has been reduced to a 19 element one-dimensional array. The relative reductions are obviously much more significant for larger features.

The Freeman chain code was adopted because it simplifies the working data set while still retaining complete information of all contour characteristics. It can make subsequent processing more manageable while still accurately representing the field of data. This fact is of considerable importance for it directly affects the real-time capability of the forecasting program. The precipitation field can vary from very simple stratiform with slowly evolving features to multicell storm environments where the data fields are complex and can change significantly between

---

14. Freeman, H. (1961) On the encoding of arbitrary geometric configurations, *IRE Trans. Electron. Comput.*, **EC-10**:260-269.

15. Wu, Li-De (1982) On the chain code of a line, *IEEE Trans. Pat. Anal. and Mach. Intel.*, **PAMI-4** (#3):347-353.

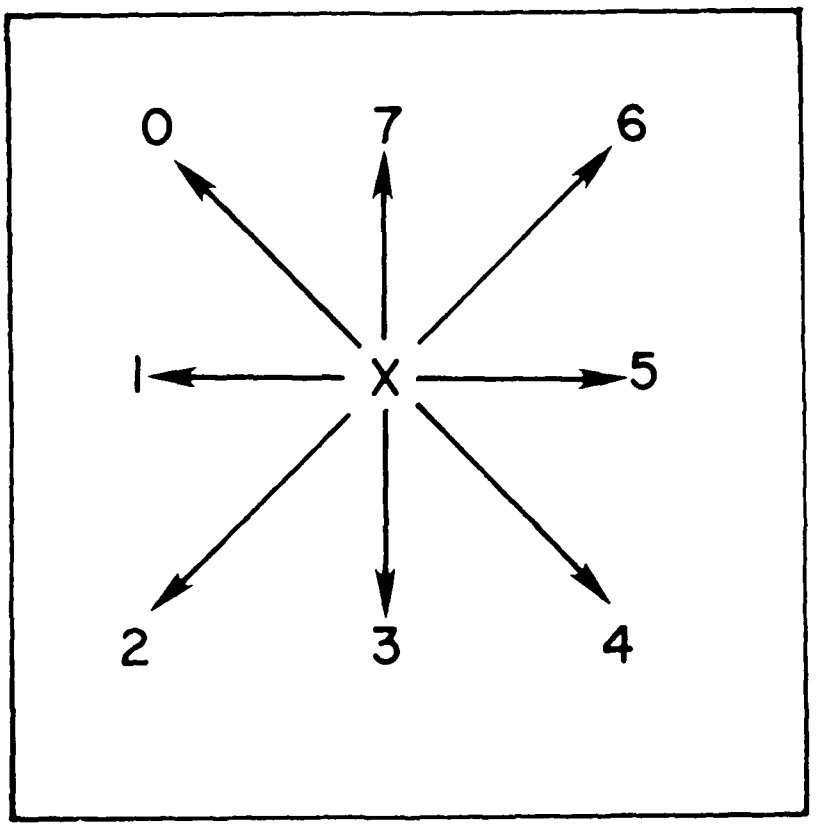

Figure 8. The eight direction vector codes used in the Freeman chain code representation

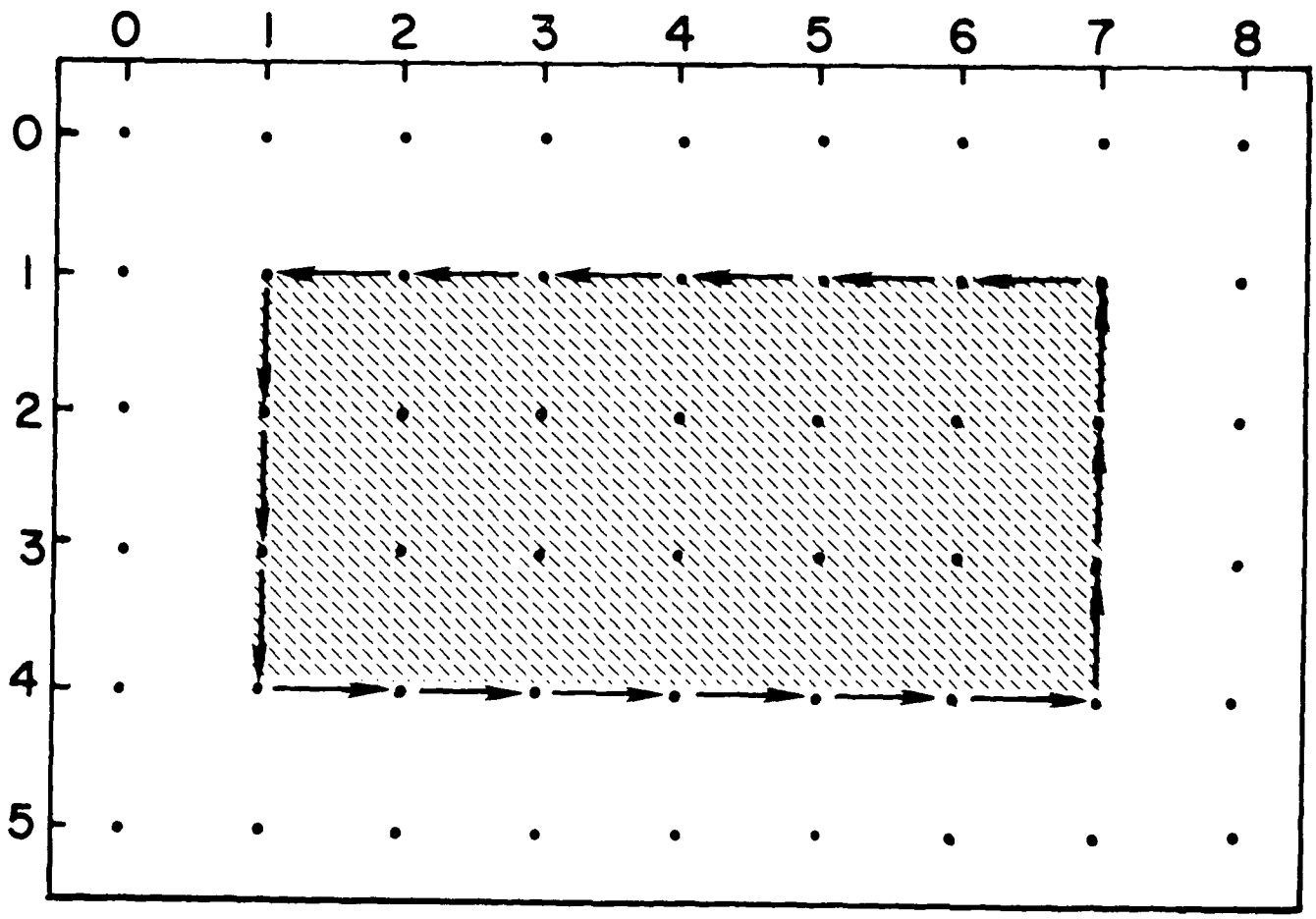

Figure 9. A rectangular region of data with a contour boundary defined by the chain code (3, 3, 3, 5, 5, 5, 5, 5, 5, 7, 7, 7, 1, 1, 1, 1, 1, 1), with origin (1, 1) and length 18

observations. Thus the methodology must be versatile while remaining representative. Further restrictions imposed by the desire to store data and perform the bulk of the calculations within the ADAGE image processor also support the use of this chain code representation. The versatility of the Freeman code will become more apparent in the following discussions of motion detection and evolution.

4. FEATURE MOTION AND EVOLUTION

4.1 Introduction to Feature Monitoring

A number of methods for monitoring the contour changes over time were investigated. Use of the chain code representation facilitates, but does not necessarily dictate, that simple analysis techniques will ensue. In fact, direct use of all chain code elements can result in a very complex and time consuming effort. Use of techniques that employ the chain code to determine "attributes" of the feature and further reduce the amount of data monitored are required and are illustrated in the following sections.

4.2 Orthogonal Vector Method

One of the first algorithms examined for mapping motions of contour points between successive observations involves the use of vectors orthogonal to the contours. For each grid point along a contour, the two neighboring chain code directions are combined to determine a resultant orthogonal vector direction. Once vector directions for all points along the contour have been determined, the contour is then aligned with the same contour at the next observation time (for example, by means of minimizing the areal difference between the two contours or by overlaying the areal centers). Orthogonal vectors are then extended from the first contour until they intersect the second. With the assumption that the intersection points are the new positions of the points from the original contour, the contour point displacements and thus complete contour change between observations, is determined.

An example of this technique is plotted in Figure 10. Quite obviously there are regions where this technique fails miserably. Where the contours are straight or strictly convex it is quite effective. However, where significant concavity or small irregularities are present many intersecting vectors are produced, resulting in obviously erroneous contour point displacements. Averaging or thinning of the orthogonal vectors to generate smoother contours or fewer contour points tends to alleviate, but not eliminate, this problem of crossover. Objectively untangling these vectors, although possible, appears too complex for real-time analysis. Thus, this technique is considered unacceptable for the current problem, but may be utilized in later efforts where highly smoothed data fields may be employed.

4.3 Contour Segment Matching

Examination of the contours in Figures 7a-d indicates that there is reasonable temporal continuity between the contours. This suggests the possibility of matching contours from one

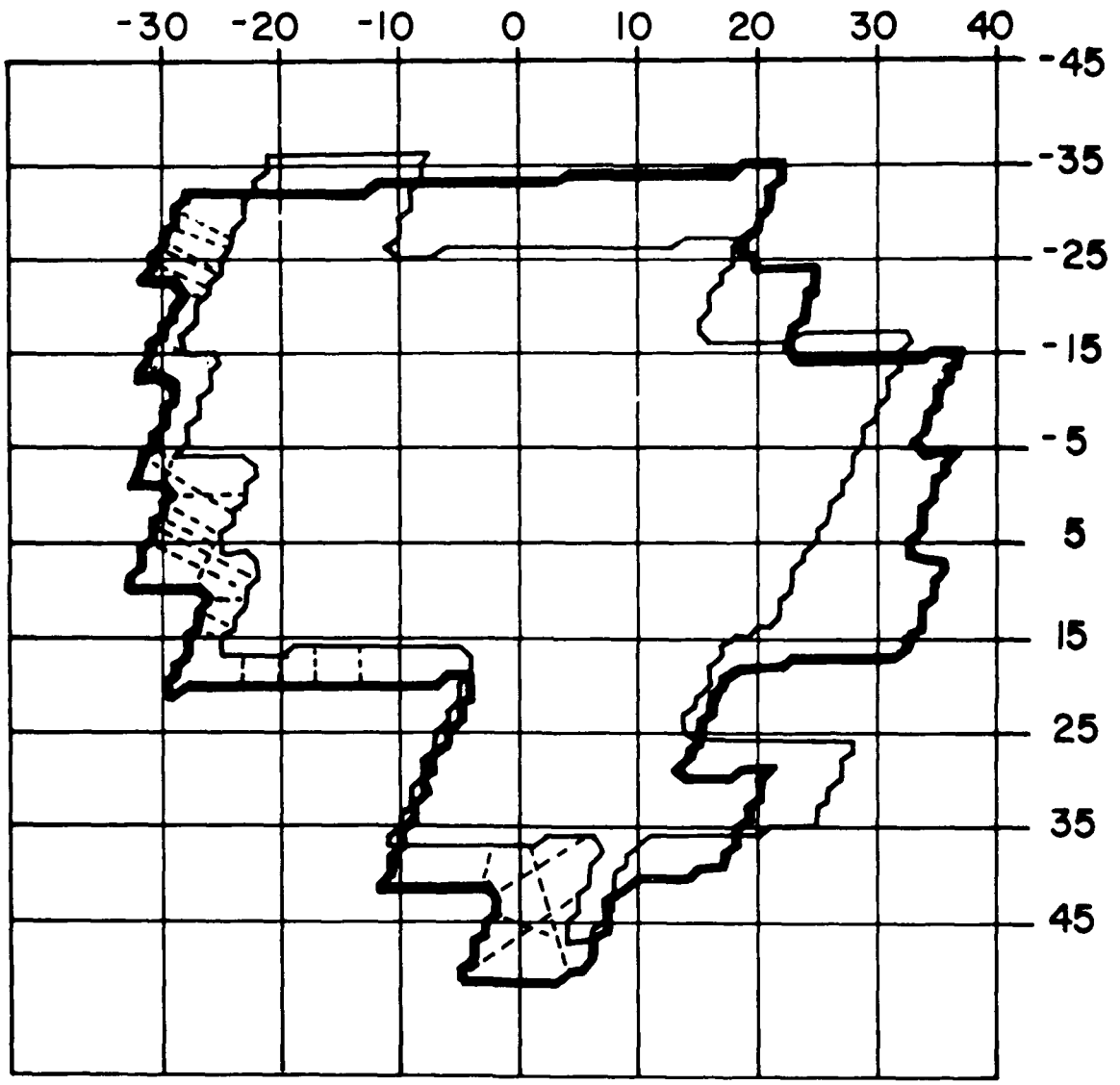

Figure 10. An example of the Orthogonal Vector Technique. Contours are of Infrared temperature for two scans 30 min apart. The heavy contour is for the earlier data. The orthogonal vectors are shown as short dashes

observation to the next. Methods employing the overlaying of two successive contours and moving them about until the areal overlap difference is minimized or collocating the centers of areas of the two contours are useful if the fields are evolving slowly. However, rapid evolution or differential motion of the contours easily disables these schemes. It is more reasonable to break up the contours into segments and then find the best matches between segments of successive contours. This method allows for monitoring the preferential growth, decay, and advection of contour segments.

Two techniques were explored for boundary segment matching:

(1) Finding the best match of straight line segments

(2) Performing a least squares fit of fixed length contour segments

Both techniques were implemented and are discussed in the next section.

4.3.1 MATCHING STRAIGHT LINE SEGMENTS

A means of classification is utilized to segment a contour chain code into a sequence of straight, or pseudo-straight, lines. Quite obviously, when working with pixel data (Cartesian data) only those lines oriented parallel to one of the axes is truly straight. Lines canted at some angle (pseudo-straight) are somewhat distorted, being composed of chain code vectors that oscillate about the mean direction. However, they can be conceptually approximated by a straight line through the mean positions. In the discussions that follow, reference to straight lines also refers to pseudo-straight lines.

The Freeman criteria for a chain code segment to be a straight line are:

(1) at most, two basic directions are present and these can only differ by 1, modulo 8.

(2) one of these values always occurs singly.

(3) successive occurrences of the principal direction occurring singly are as uniformly spaced as possible.

Application of all three criteria in the operational arena is overly restrictive and results in a multitude of very short line segments that become difficult to track between observations. Therefore, conditions were relaxed by invoking only the first criterion. This reduces the number of computations, tends to decrease the total number of segments, and accordingly increases the size of some segments.

Beginning with the first chain code element the subsequent codes are surveyed so as to find the longest segment including that first element that conforms to the definition of a straight line. Once identified, the beginning segment number, length, and primary direction of the line are recorded. This process is done iteratively, each new starting point being the first contour point after the end of the previously determined line segment. When the absolute starting point is included in a line segment the last line segment has been determined and the segmentation process stops. For example, the directional code for the box in Figure 9 is given by

C = 3, 3, 3, 5, 5, 5, 5, 5, 5, 7, 7, 7, 1, 1, 1, 1, 1, 1

The maximum line lengths for each direction can be easily identified here and are shown in Table 1. In real situations, there are many more straight line segments, each identified by their direction code, length in terms of number of direction codes in the line, and the starting position of the line relative to its location from the absolute beginning of the contour chain code. The listing shown in Table 1 is in

Table 1. Listing of Freeman line code attributes for the rectangular areas of Figures 9 and 11. Ranking is first by length followed by absolute starting position.

FEATURE FIGURE 9				FEATURE FIGURE 11			
LINE No	START	LENGTH	DIRECTION	LINE No	START	LENGTH	DIRECTION
1	3	5	5	1	10	7	1
2	11	5	1	2	3	4	5
3	0	3	3	3	0	3	3
4	8	3	7	4	7	3	6

the order determined by the straight-line algorithm which ranks first by length and second by absolute segment starting location.

Now assume that the feature observed in Figure 9 was observed later (Figure 11) with the resultant chain code:

C = 3, 3, 3, 5, 5, 5, 5, 6, 6, 6, 1, 1, 1, 1, 1, 1, 1

The results from the straight-line analysis are also included in Table 1. Due to evolution, the results for the two examples are somewhat different. With the straight line segments for a given contour identified for two successive data fields, the process of matching segments between observations is begun.

It must be noted that a contour is continuous and the starting location is basically arbitrary. In the example here, the upper-left-most pixel is defined as the origin, but it could just as well have been the lower-right-most. Therefore, the code and associated data are treated not as an ordered linear list, but in a circular fashion, for example, a circular queue. Since the chain code is a directional code the shape and orientation of the contour are always preserved, independent of the absolute starting location of the contour code.

The rule hierarchy that has shown the greatest success is:

(1) Search for the longest segments in each contour and match the segments that have the same code.

(2) Proceed iteratively with successively shorter segments until no more matches can be made.

(3) For those remaining line segments attempt to find approximate matches by comparing starting locations.

For the simple examples of Table 1, the matchups would be line 1 to 2, line 2 to 1, line 3 to 3 and line 4 to 4.

4.3.2 CURVE FITTING CHAIN CODE SEGMENTS

The second approach in curve fitting is to break down the contour chain code into segments having a fixed number of elements regardless of chain code values. This technique focuses on matching similar shapes, a method of pattern recognition. While the straight-line segment approach adopts the simple technique of matching segment lengths and code values, here a more complex procedure is required: namely, least-squares fitting of the chain code directions between segments in successive observations. The underlying assumption is that between observations the processes of advection and evolution still allow contour segments to retain their overall shapes. The chain code of the selected contour is divided into segments of specified length, usually 10 to 25 percent of the total number of codes in the chain. Each one of these segments is then compared with every possible segment of identical length in the contour chain code set of the next observation. For each comparison, the mean difference and the mean square of the differences of the individual chain code directions are computed regardless of location along the contour. A perfect match is obtained when the mean difference is zero and the mean square of the differences is a minimum.

An example of this process is shown in Figure 12 where these two parameters are plotted against starting segment number for the second contour. There is a very obvious minimum of the mean

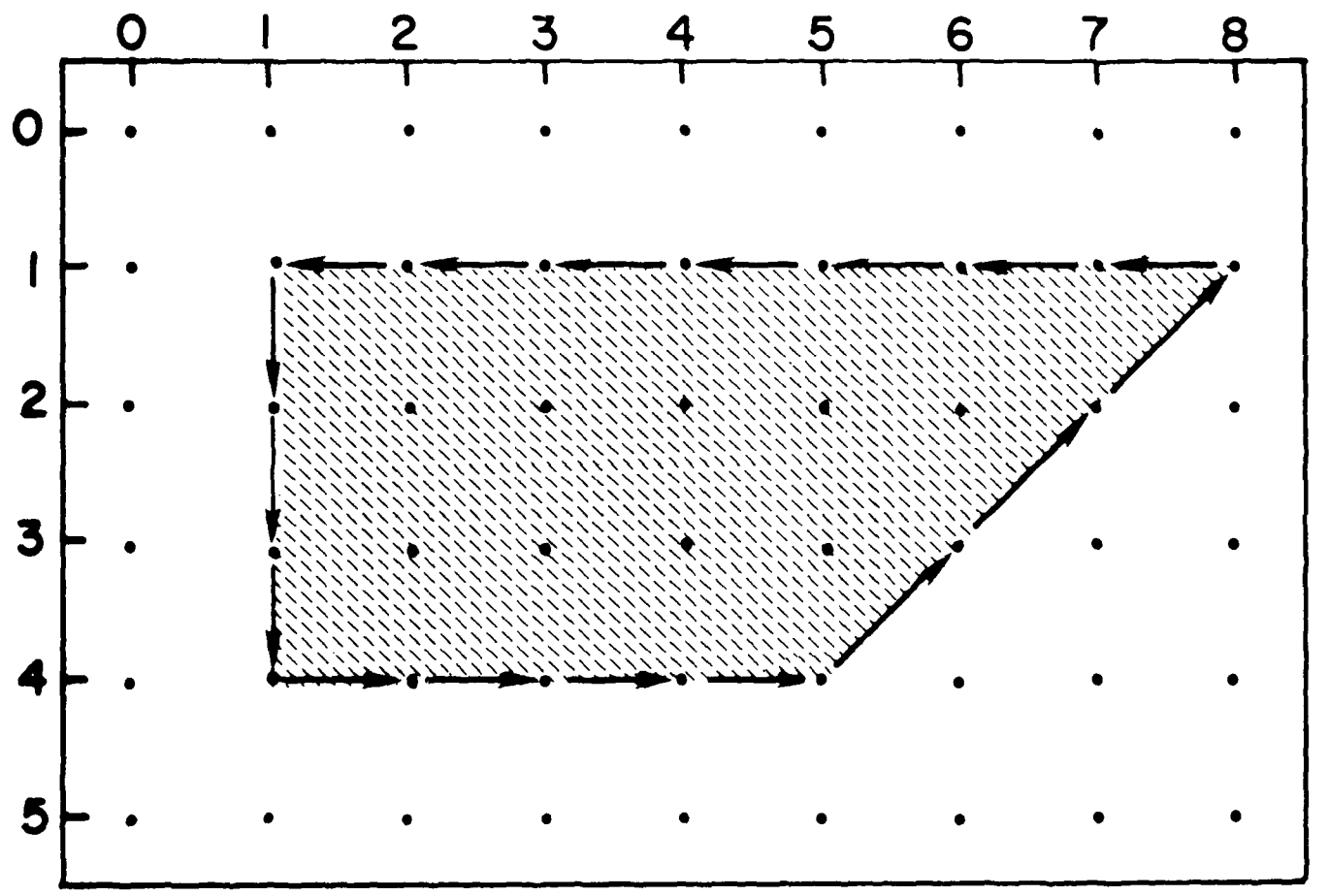

Figure 11. Evolving rectangular contour of Figure 9 observed at a later time

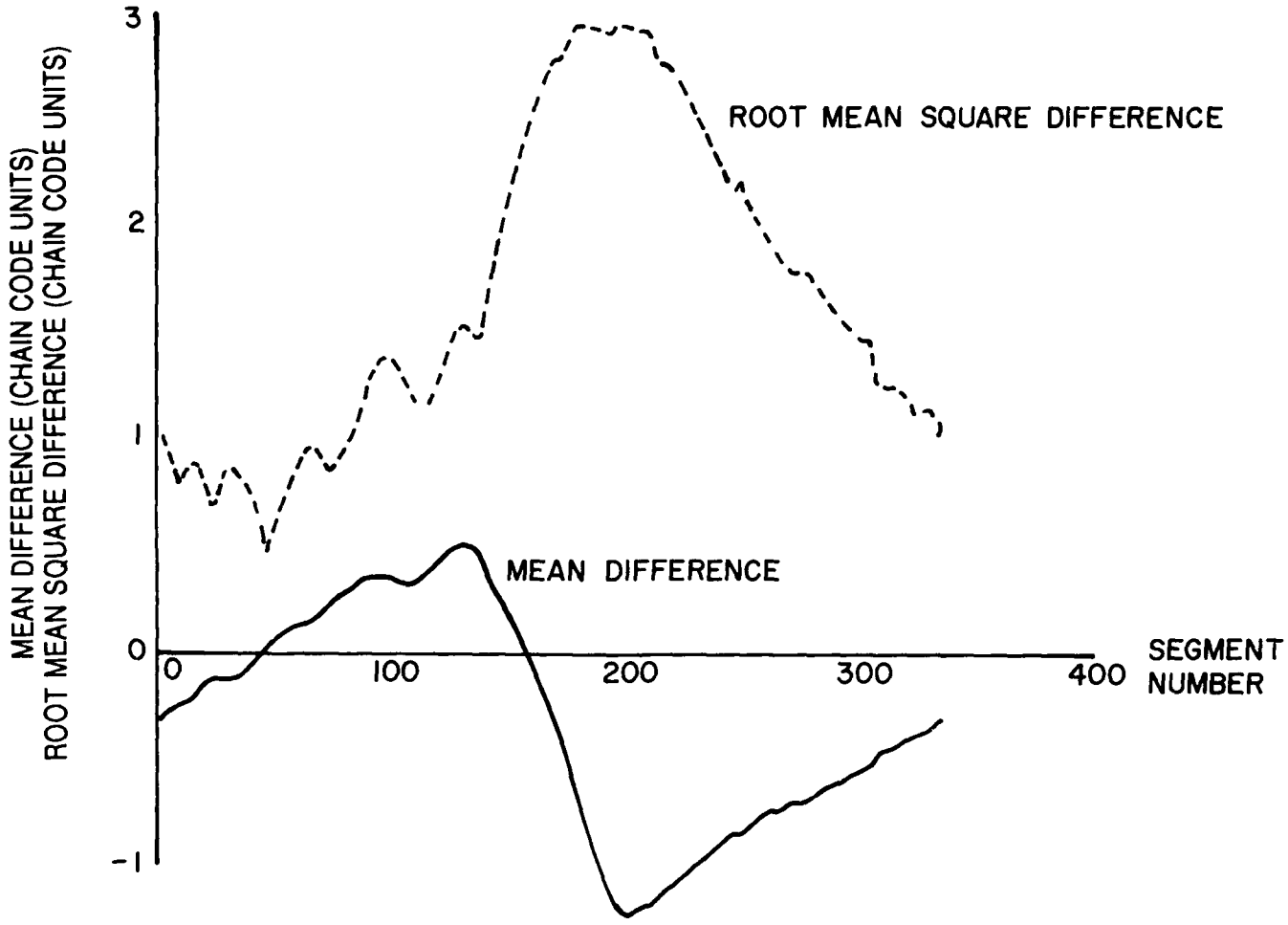

Figure 12. Plot of mean difference (solid) and mean square difference (dashed) measurements resulting from curve fitting a contour segment between two successive times

squares of the differences at segment position 308 along with a near zero of the mean difference. This combination represents the best match for this case. Note that the crossover of the mean difference curve from negative to positive near segment location 175 occurs with a maximum of mean square difference, and thus is not a good match. Not all attempts at curve matching produce such pleasing results, particularly where there is considerable evolution between observations. However, in all cases where the fit looks reasonable to the eye, the best fit is usually identified with the minimum mean square difference. Thus, the resulting matching criteria are a near minimum mean square combined with a relatively small mean difference.

4.4 Attribute Determination and Evolution Detection

Both the straight line and curve fitting algorithms perform mapping between segments of contours from successive observations. Each segment has several associated characteristics; including a segment length, orientation, starting point value, and distance from some point of reference to a given point in the segment. This limited set of quantities completely describe the contour segments and are termed the segment "attributes". If there is reasonable continuity between successive contour observations, that is, if segment matching between the contours may be accomplished, then one has a mechanism for monitoring the evolution and motion of contour segments and ultimately the contours themselves. Through use of relatively simple techniques, the two-dimensional data fields can be described in terms of selected contours. These contours can be subdivided into contour segments, which can then be described in terms of segment attributes. It is only necessary to monitor and forecast attributes to develop forecasts of future cloud and precipitation field distributions. Thus, it is segment attribute histories that will serve as input to the forecasting algorithms.

5. FORECASTING

5.1 Forecasting Considerations

A number of restrictions are imposed upon the forecasting process. First, the desire to develop forecasts in real-time demands techniques that run quickly and efficiently. Second, the frequently short lifetimes of some storms require that forecasts be developed from limited data histories. Third, the desire to store all necessary historical data and perform the bulk of the data processing within the image processor demands use of analysis techniques that minimize data storage requirements. Computer memory and time constraints also demand that the number of quantities input to the forecasting process be as small as possible, while still adequately defining the evolving nature of the storm precipitation intensity contours. These considerations have profoundly influenced the development of both the data storage and forecasting methods.

These considerations, as previously discussed, have led to tracking and forecasting a highly reduced set of variables termed contour attributes. These have been identified to include such items as line length, location, and orientation, and chain segment orientation and location. Forecasting the evolution of a contour thus requires monitoring these attribute values for all contour segments,

developing the historical attribute data sets, forecasting the future attribute values, and reconstructing and linking the forecast contour chain code segments.

The attributes being followed and forecast are not standard meteorological quantities, but rather a very limited set of arbitrarily defined variables. The forecasting process may thus be approached from a nonmeteorological orientation, at least in the sense that (1) there is no physical model for providing a priori expectation of the behavior of the attributes and (2) attribute variations may not easily be linked to other observations of the physical environment. It is assumed that time variation lies somewhere between a state of stationarity to perhaps quadratic variation. For example, if the line length or orientation for a contour segment were stationary, then its value would remain constant during observations. If a linear variation were observed in line length of a non-rotating line, then the total number of code elements composing the line would be increasing or decreasing by a constant rate between observations. This assumption can be severely strained during periods of rapid storm development where production of new cloud and precipitation may occur on the order of minutes. However, rapidly evolving storm environments where only a very limited number of observations are available make estimation of more complex time trends (for example, quadratic) grossly inaccurate. Thus, the assumption is made that the attributes are random quantities, a combination of some quasi-linear trend variation and noise.

The forecasting process has focused on extrapolation routines that employ smoothing functions. These routines either explicitly or implicitly employ knowledge of past attribute variation. Some provide for user-selected preferential weighting of current or past observations, or automatic adaptation to the changing environment through monitoring forecast errors. The routines also perform automatic filtering of noise contributions. Finally, they are hig r efficient, requiring only a very limited amount of input information and computer operations to generate new forecast values. The following material will address some relevant features of these classes of routines and present some observations of their utility and deficiencies, leading to the selection of preferred forecasting algorithms.

5.2 Forecasting Techniques

5.2.1 TECHNIQUE FORMULATIONS

A total of nine forecasting techniques were considered, providing an overall capability for assimilating data ranging from stationary to quadratic behavior. The routines are listed in order of computational complexity in Table 2, and the generic formulations describing them are presented in Table 3. Before discussing their performance with data, it is instructive to develop a clearer understanding of the characteristics of these classes of routines.

The stationary formulations, as indicated in Tables 2 and 3, inherently assume the variable being forecast is statistically stationary (for example, in time), perhaps a combination of a constant mean value with an added random component. The Simple Moving Average (SIMMOVAV) method is a moving box filter, using the average of the latest three observations as the forecast value. The Simple Exponential Filter (SIMEXP) method develops a forecast from a weighted sum of the current observation and the previous forecast for the current time and employs a user-selected constant weighting factor allowing for preferential weighting of the two terms. The Adaptive Simple

Table 2. Forecasting techniques tested for use in RAPID ranked by computational complexity.

FORECASTING TECHNIQUES

NO.	ROUTINE NAME	TITLE	NUMBER OF WEIGHTS FIXED	NUMBER OF WEIGHTS ADAPTIVE
		STATIONARY ROUTINES		
1.	SIMPLE MOVING AVERAGE	(SIMMOVAV)	0	0
2.	SIMPLE EXPONENTIAL	(SIMEXP)	1	0
3.	ADAPTIVE SIMPLE EXPONENTIAL	(ADSIMEXP)	0	1
4.	MODIFIED KALMAN	(KALMAN)	0	1
		LINEAR TREND ROUTINES		
5.	BROWN ADAPTIVE 1 PARAMETER	(BWNIPAD)	0	1
6.	LINEAR MOVING AVERAGE	(LINMOVAV)	0	0
7.	HOLT 2 PARAMETER	(HOLT2P)	2	0
8.	LINEAR REGRESSION	(LINREG)	0	0
		QUADRATIC TREND ROUTINES		
9.	BROWN QUADRATIC EXPONENTIAL	(BWNQUAD)	2	0

Table 3. Generic formulations of the forecasting techniques listed in Table 2 ranked by computational complexity.

FORECAST FORMULATION	TECHNIQUE NO.
$F(t+m) = (X(t) + X(t-1) + \ldots X(t-N+1))/N$	1
$F(t+1) = WEIGHT \star X(t) + (1-WEIGHT) \star F(t)$	2,4
$F(t+1) = WEIGHT(t) \star X(t) + (1-WEIGHT(t)) \star F(t)$	3
$F(t+m) = A(t) + B(t) \star m$	5-8
$F(t+m) = A(t) + B(t) \star m + .5 \star C(t) \star m \star m$	9

(X(t) = CURRENT OBSERVATION)

(F(t+m) = FORECAST FOR NEXT OBSERVATION TIME)

Exponential (ADSIMEXP) method uses the same formulation as SIMEXP but it allows the relative weighting of current and historical data to adjust automatically with each new observation. This weighting factor is determined from the ratio of total forecast error to total absolute forecast error, where error is defined as the difference between the past forecast for the current time and the current observation (error = F(t) - X(t)). Since the numerator (total error) may fluctuate between positive and negative values, and the denominator (total absolute error) continually increases, the magnitude of the weight decreases with increasing time. Thus the asymptotic trend is for diminishing dependence upon current observations and growing dependence upon the historical mean value. As more data are acquired the historical mean will approach the population mean (if observations are in fact a sum of mean and noise terms). The Modified Kalman (KALMAN) filter works similarly to the standard Kalman filters except that it employs only the latest three observations and forecasts in determining the new weights. In this modified form the automatically adjusting weight applied to the current observation is equal to the variance of the last three forecasts, divided by the sum of the last three variances of the observations and forecasts.

While the SIMMOVAV (moving box filter) method employs only the latest three observations in deriving a new forecast, the SIMEXP, ADSIMEXP, and KALMAN methods implicitly utilize the current and all previous observations, however, old observations become increasingly less important as new observations are acquired. This dependence upon historical data can be easily shown for the SIMEXP method:

$$
\begin{aligned}
F(t) &= A*X(t) + (1-A)*F(t) \\
&= A*X(t) + (1-A)*(A*X(t-1) + (1-A)*F(t-1)) . \\
&\quad \vdots \\
&= A*X(t) + A*(1-A)*X(t-1) + \ldots A*(1-A)**(N-1)*X(t-N-1)
\end{aligned}
$$

Except for SIMMOVAV the underlying concept remains the same, namely: (1) the forecasts are weighted sums of all observations; and (2) recent observations are weighted more heavily than old observations.

The nonstationary formulations tested fit either a linear trend or quadratic variation to the data. The Linear Moving Average (LINMOVAV) filter employs two quantities: (1) a weighted mean of the past N observations (MEANOBS(N) = (obs(1) + obs(2) +...obs(N))/N) , and (2) a similarly weighted mean of the past N mean values (MEAN = MEANOBS(1) + MEANOBS(2) + ...MEANOBS(N))/N). The linear Adaptive 1-Parameter Brown (BWN1PAD) method revises both the origin and slope parameters through use of a correction factor determined from the difference between the previous forecast and the current observation. The linear Holt 2-Parameter (HOLT2P) method performs two smoothing operations and revises the origin and slope terms independently with differing user-selected weights. The Linear Regression (LINREG) method is of standard form. It explicitly weights all data equally. Current data storage requirements restrict this technique to a maximum of the ten most recent observations. The Brown Quadratic Exponential (BWNQUAD) formulation employs three averaging actions with a single user-selected weighting factor: namely (1) the average of the observations; (2) the average of the mean observations; and (3) the average of the average of the mean observation. Although it appears somewhat convoluted, this method provides a relatively simple way of estimating

first and second order trends, while at the same time filtering out random noise contributions. Except for the LINMOVAV routine, these methods also employ all historical data.

5.2.2 TECHNIQUE INITIALIZATION AND USER OPTIONS

Since these routines are designed to employ a number of observations in deriving any individual forecast, they generally cannot be "turned on" with acquisition of the first or second observation. Rather, the routines employ initialization procedures that allow for generation of forecasts through use of alternative and simpler methods until sufficient data are acquired to run the exact formulations. These initialization procedures and the time delays before the first forecasts are available through use of the exact formulations are presented in Table 4.

In initializing the stationary techniques it is assumed that the first observation is a good estimate of the population mean and that it can be used as a first guess forecast. Thus, use of the stationary data routines allows one to develop a forecast after the first observation. Use of "difference" data, which are the differences between successive observations, delays the forecasting process by one observation period. As a result, the first forecast is obtained with acquisition of the first "difference" data measurement, or second data observation. The time to pass through the initialization stage and enter the true formulation varies between routines. For example the SIMMOVAV routine, which performs a running average of the latest three observations, does not employ the exact formulation until the third observation period. On the other hand, the SIMEXP routine, which employs the current observation and the previous forecast, engages the exact formulation with acquisition of the second observation. To initialize the nonstationary data routines, it is assumed that the first two observations represent a good first estimate of any linear trend present in the data. The various initialization procedures may obviously be suspect in environments where rapid evolution is underway, or where the observations include significant noise contributions. However, the lifetimes of some meteorological events can be quite short and initialization is necessary for rapid forecasting of future attribute values.

The delay before entry into the exact formulation can be controlled for some techniques (for example, SIMMOVAV, KALMAN, LINMOVAV) through selection of the number of entries utilized in the exact formulation. The requirement that routines be responsive to new trends in the data translates to the employment of only a limited number of the most recent observations. Two observations provide an estimate of change in an attribute value. However, if the noise contributes significantly to the observed change, then forecasts employing trend estimation tend to be in great error. A greater number of observations provide additional stability in the trend estimation process. The combination of potentially short feature lifetimes and the desire to develop forecasts quickly resulted in the selection of three data values to drive the exact formulations where user-selection is allowed.

The selection of optimum weights for the various routines is determined through observation of their performance in differing data environments. One would expect an advantage in using adaptive methods where the weights are adjusted automatically, with the adjustment generally being dependent upon the error between the observed and forecast values. The nine routines were evaluated on a series of test data, both simulated and real, to characterize their behavior and ascertain the optimum weight values.

Table 4. Procedures for initializing forecasting techniques listed in Table 2. Technique may not develop (none), or may use first observation (obs(1)) as, a first forecast after first observation. Exact formulation engagement noted by ALGORITHM.

FORECAST INITIALIZATION PROCEDURES

TECHNIQUE		FORECAST FORMULATION FOR OBSERVATION NUMBER		
NO.	NAME	1	2	3
1.	SIMMOVAV	obs(1)	(obs (1) + obs (2))/2	ALGORITHM
2.	SIMEXP	obs(1)	ALGORITHM	
3.	ADSIMEXP	obs(1)	(obs (1) + obs (2))/2	ALGORITHM
4.	KALMAN	obs(1)	(obs (1) + obs (2))/2	ALGORITHM
5.	BWNIPAD	none	(obs (2) + obs (2) – obs (1) ★ m	ALGORITHM
6.	LINMOVAV	none	(obs (2) + obs (2) – obs (1) ★ m	ALGORITHM
7.	HOLT2T	none	(obs (2) + obs (2) – obs (1) ★ m	ALGORITHM
8.	LINREG	none	ALGORITHM	
9.	BWNQUAD	none	none	ALGORITHM

5.3 Forecast Technique Evaluation with Test Data

5.3.1 GENERATED TEST DATA

A small number of simple, but illustrative, data sets were generated for testing the forecasting routines. The three data sets, as shown in Figure 13, consist of: (1) a constant mean with oscillation about the mean; (2) a pure linear trend; and (3) a pure quadratic trend. A real-world scenario was assumed where only a limited number of observations were made available to the routines. Although these data did not allow for the effects of random noise, the relative performance of the methods in the controlled forecasting environment provide insight into the effects of initialization and their general response. Figure 13a is representative of data having constant mean, but with an added error (oscillation) component. Two data sets are shown, one starting at zero, and the second starting below zero. The second set was employed primarily to observe the behavior of nonstationary data routines when the initial data erroneously suggested a trend was present in the overall data set. Figure 13b shows the purely linear data set and Figure 13c the purely quadratic.

A number of factors were considered in comparing the relative performance of the various routines, including: (1) actual forecast error; (2) total accumulated forecast error; (3) total absolute forecast error; and (4) rate of response to changes in data character. These quantities were determined for one-interval, two-interval, and three-interval forecasts. An interval is the time between the start of two successive observation sequences. Assuming new observations are acquired every 8-10 minutes, this would produce the desired forecast warning times of 0 - 0.5 hr. Discussions will first focus on stationary routines, then later be directed towards nonstationary routines. For those methods where user-selected weighting was available the three weights 0.3, 0.5, and 0.7 were employed. The plots shown will present the results from the cases which exhibited minimum error.

Samples of actual forecast error obtained with stationary data routines are presented in Figures 14a-c. The range of actual error incurred generally lies between +/-0.2, comparable to the variation of the input data about the population mean. The initialization procedures all allow a first forecast to be generated with the first observation and the first measure of forecast error is obtained with the second observation. Since all the stationary data routines are initialized similarly, they all produce the same first forecast and incur the same initial error. Differences between routines become apparent with the third observation, when the exact formulations are in force. These results indicate that all routines perform quite similarly, the only significant exception being ADSIMEXP which responds more quickly to input data variation than the other stationary routines. ADSIMEXP was found to overreact to data variation with all test data sets. The oscillation in forecast error simply reflects forecasted values that also oscillate, but which lag the input data sequence. The KALMAN method shows negative (< 0) bias indicating the forecast underestimate is greater than the magnitude of its overestimate. The other routines produce forecast underestimates and overestimates of nearly equal magnitude.

The accumulated forecast error is useful for indicating the presence of biases in the forecasts and are shown in Figures 15a-c for the stationary routines. The initial positive (> 0) biases result from the first observation being greater than the second, causing forecast overestimates. The error values oscillate about zero and show no particular trends except for the KALMAN method, which has a trend towards larger negative values. The results for the second data set (start below zero) are very similar to those shown here, except that the accumulated errors are all displaced downward by about 0.4. This

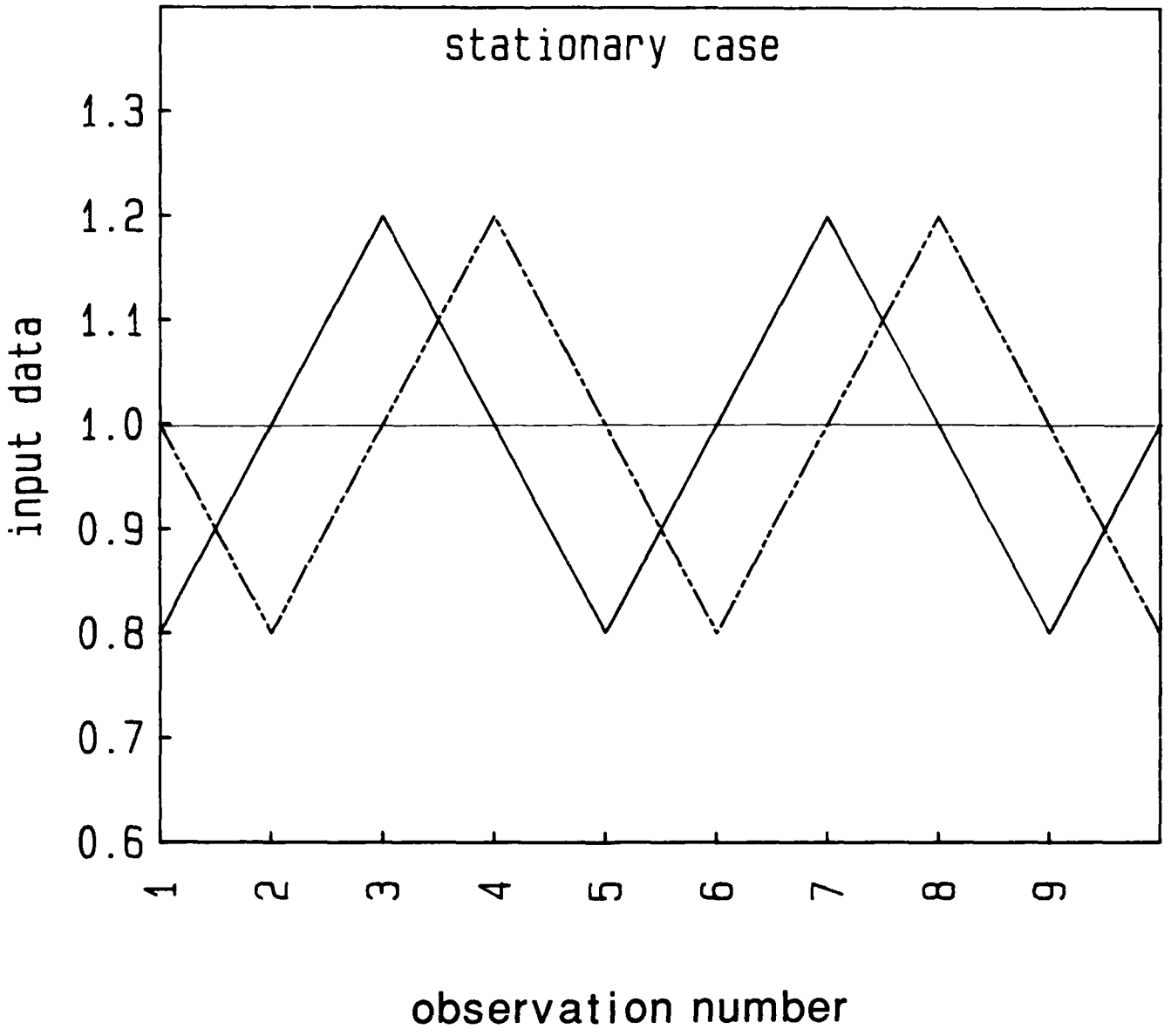

Figure 13a. Generated test case data for evaluating forecasting routines: stationary data with zero (dashed) and negative (solid) starting values

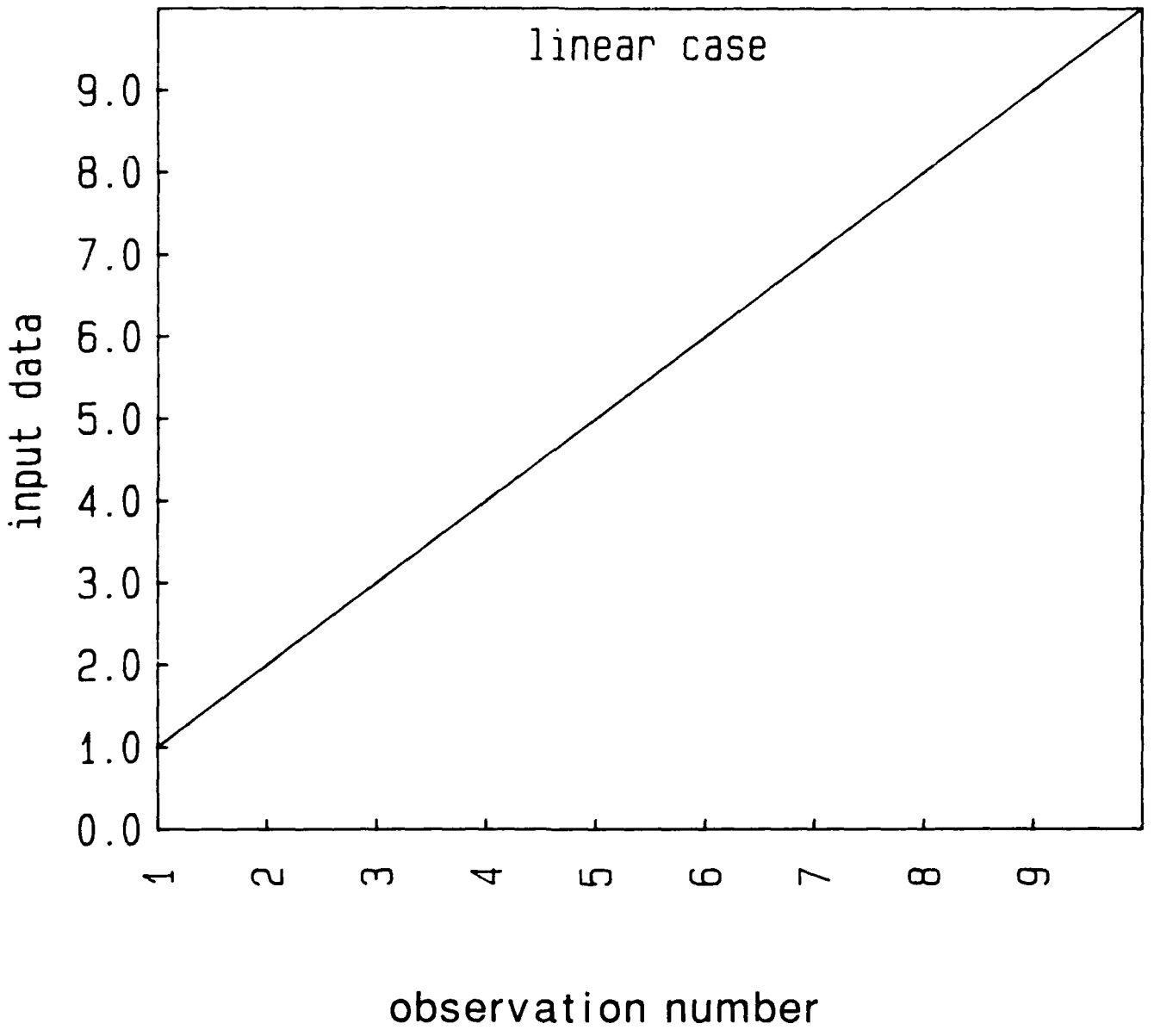

Figure 13b. Generated test case data for evaluating forecasting routines: pure linear

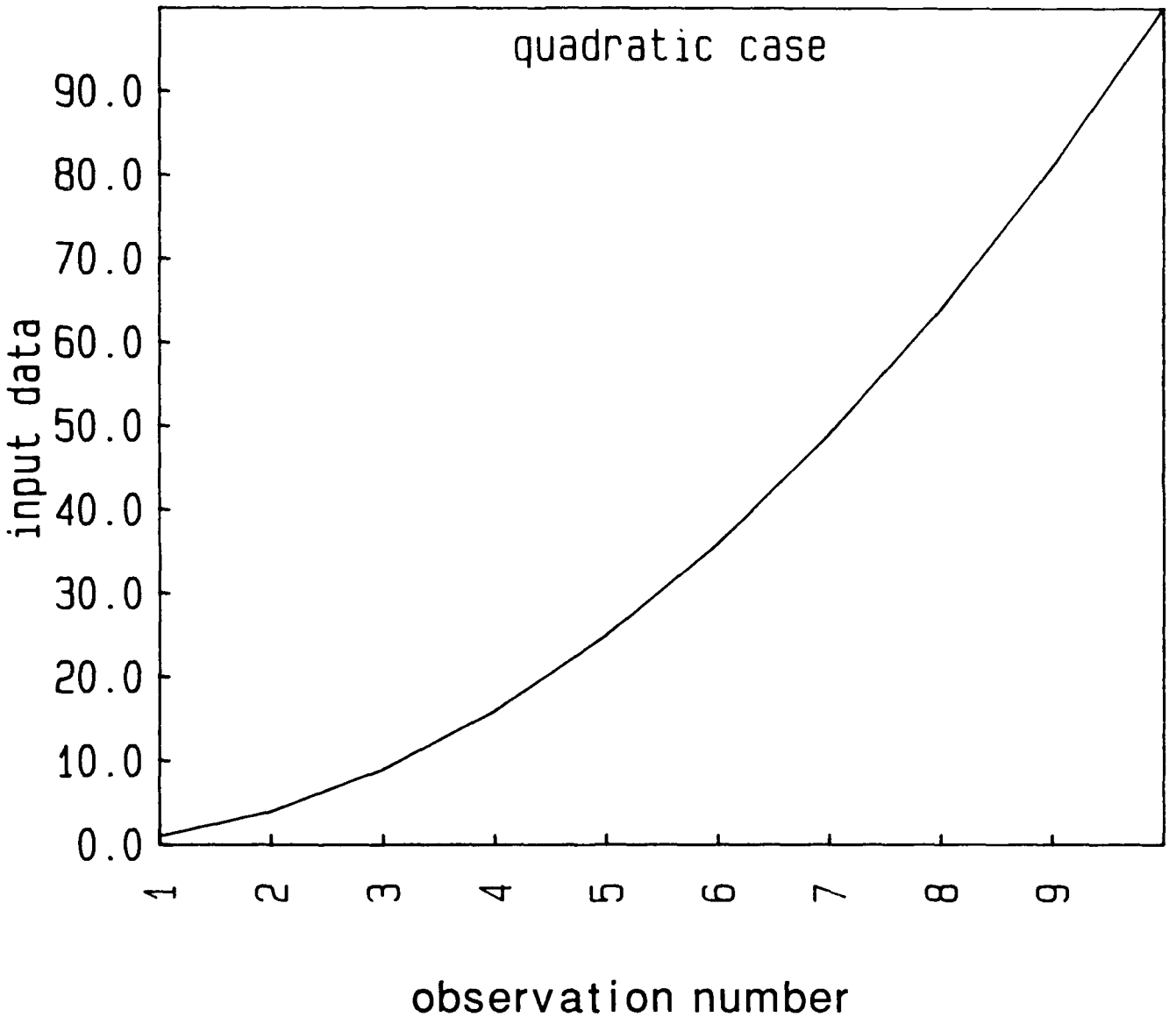

Figure 13c. Generated test case data for evaluating forecasting routines: pure quadratic cases

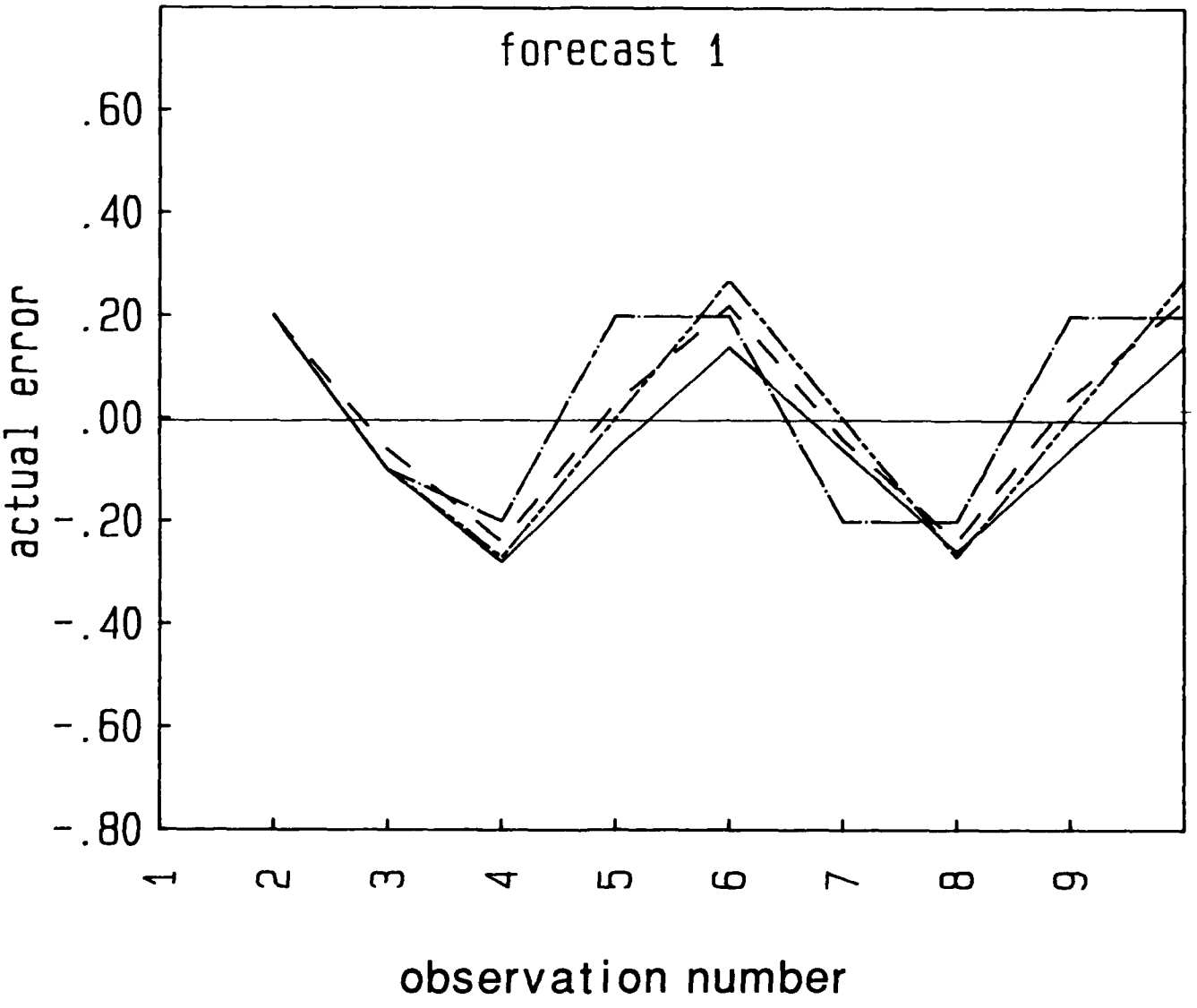

Figure 14a. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using stationary data (zero start) for one-interval forecasts

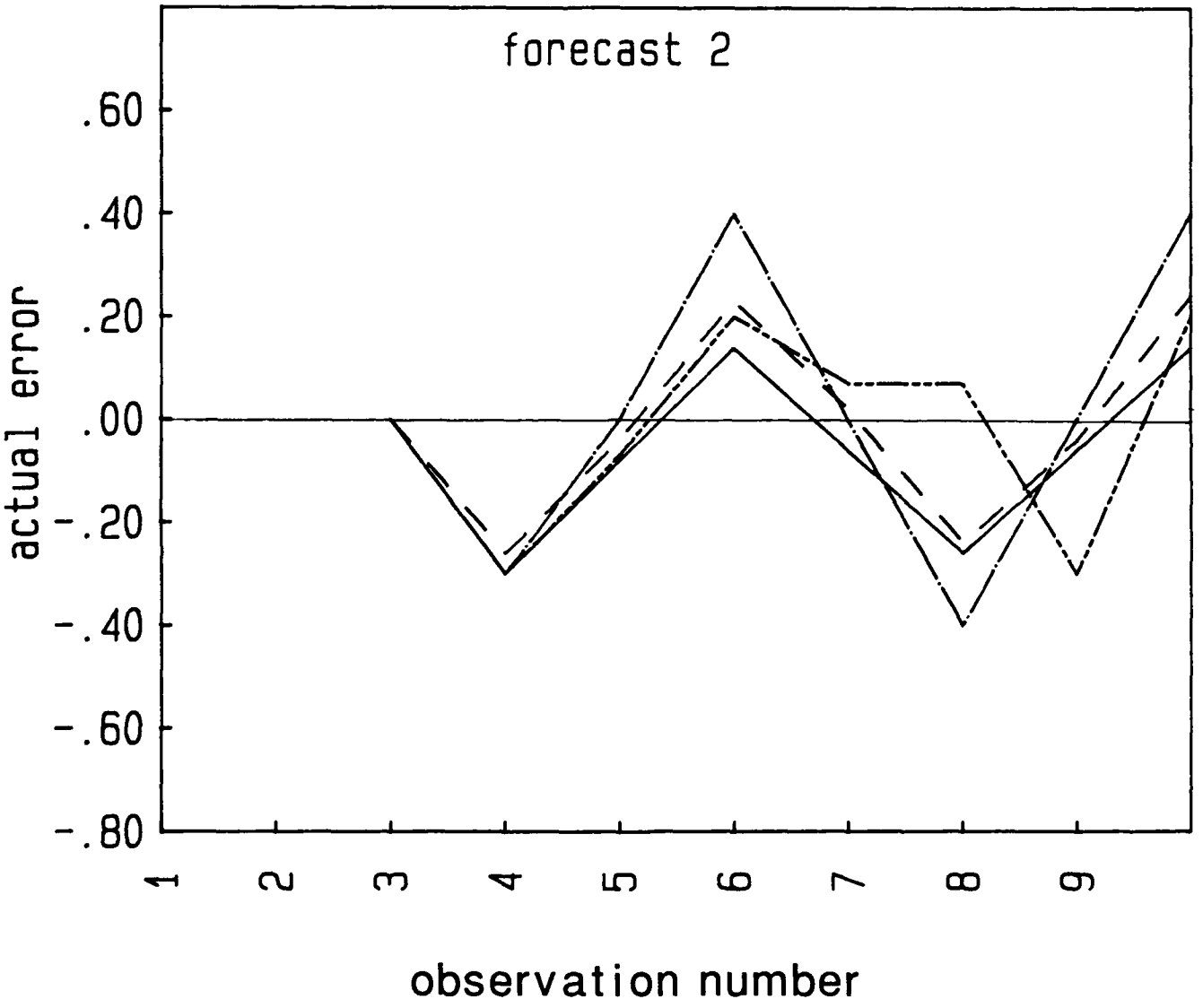

Figure 14b. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using stationary data (zero start) for two-interval forecasts

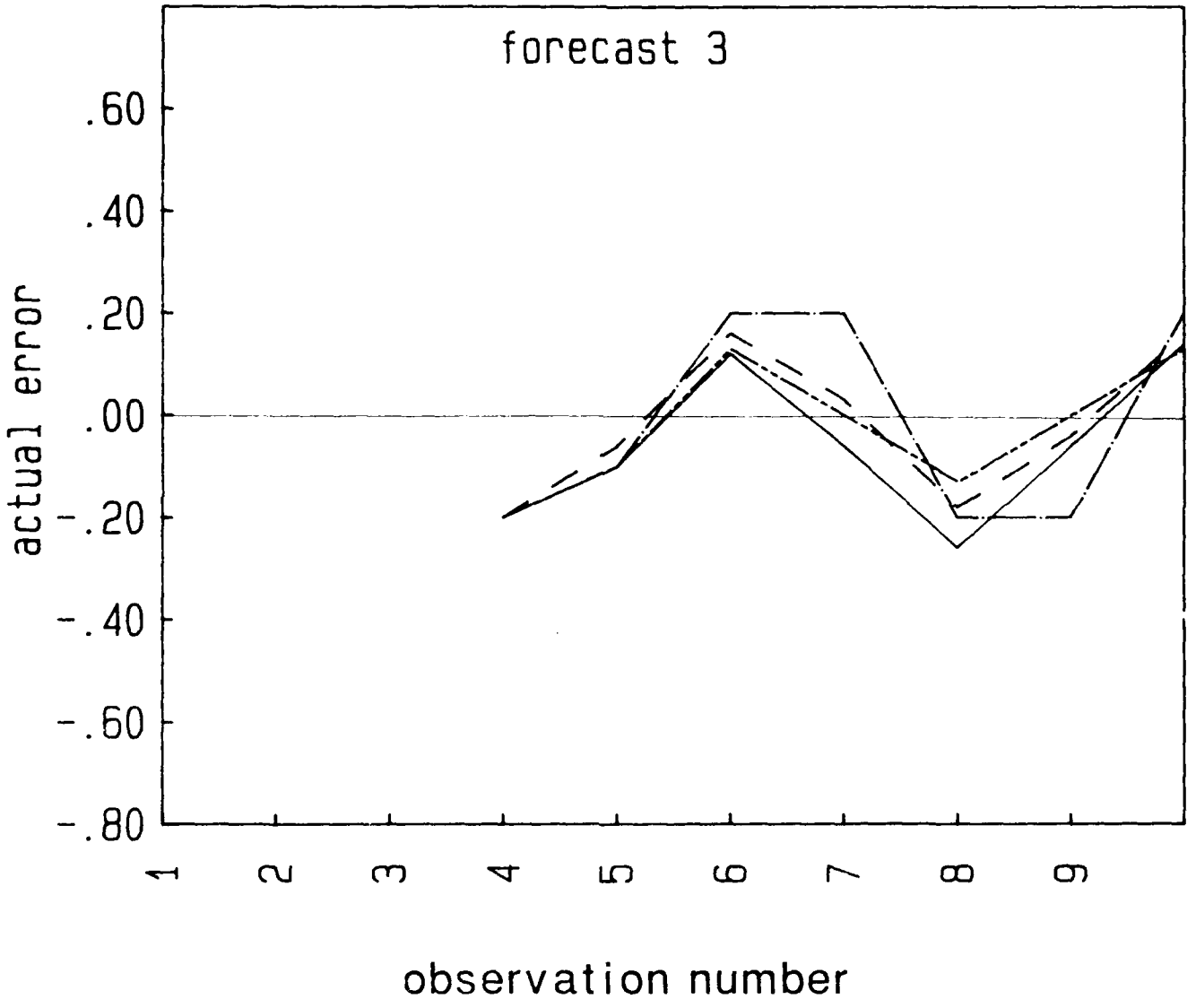

Figure 14c. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using stationary data (zero start) for three-interval forecasts

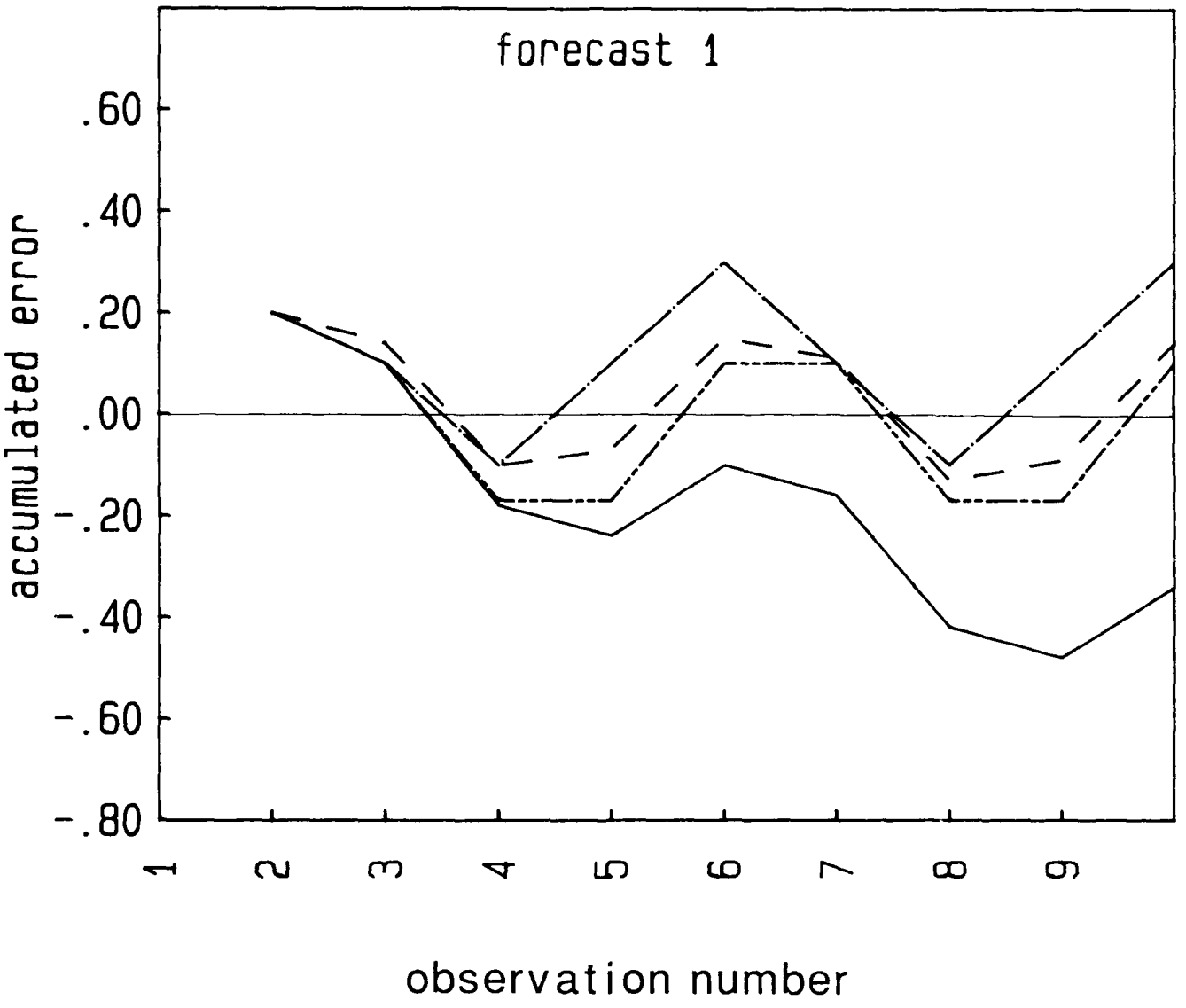

Figure 15a. Plots of accumulated forecast error corresponding to Figure 14 data

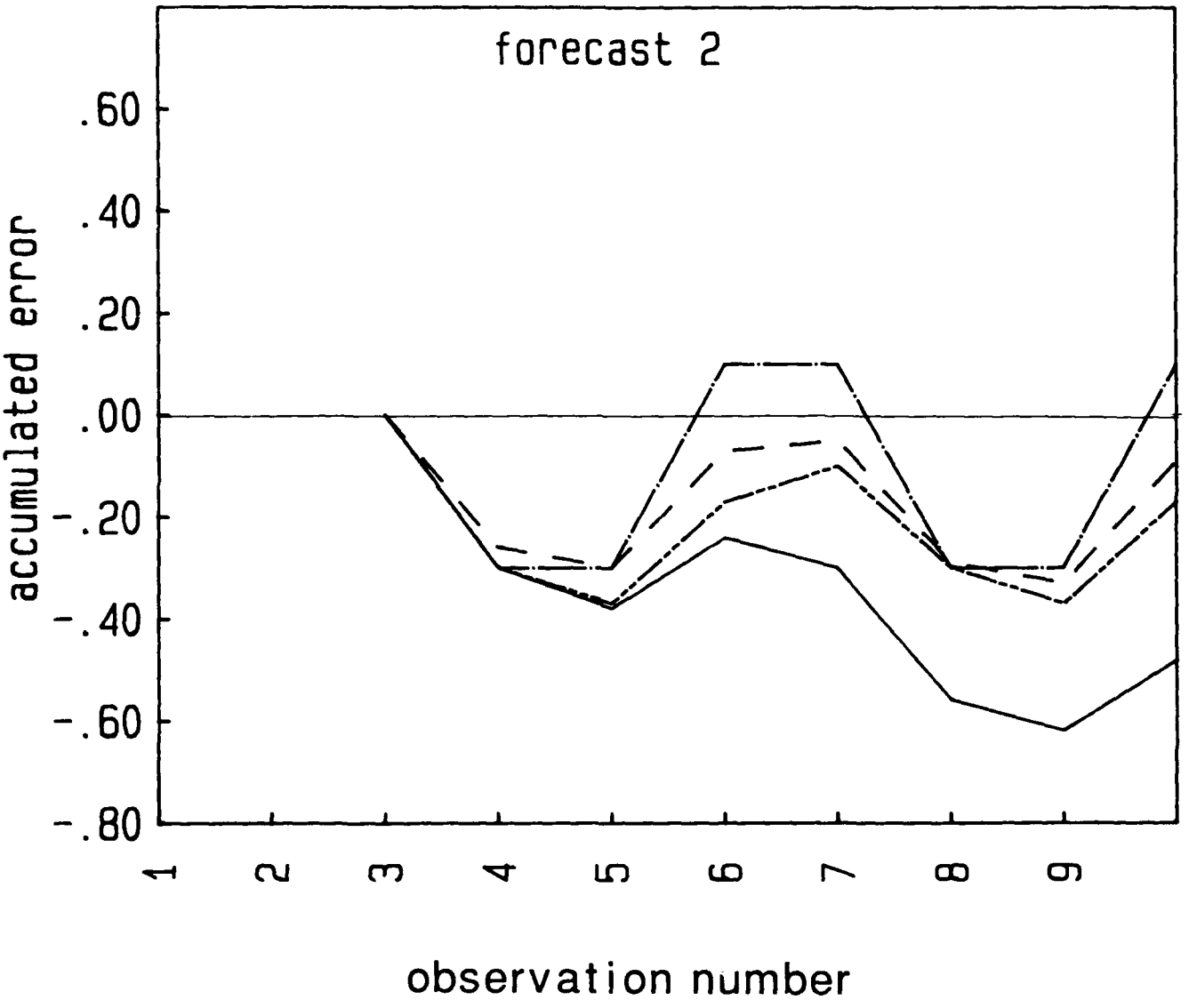

Figure 15b. Plots of accumulated forecast error corresponding to Figure 14 data

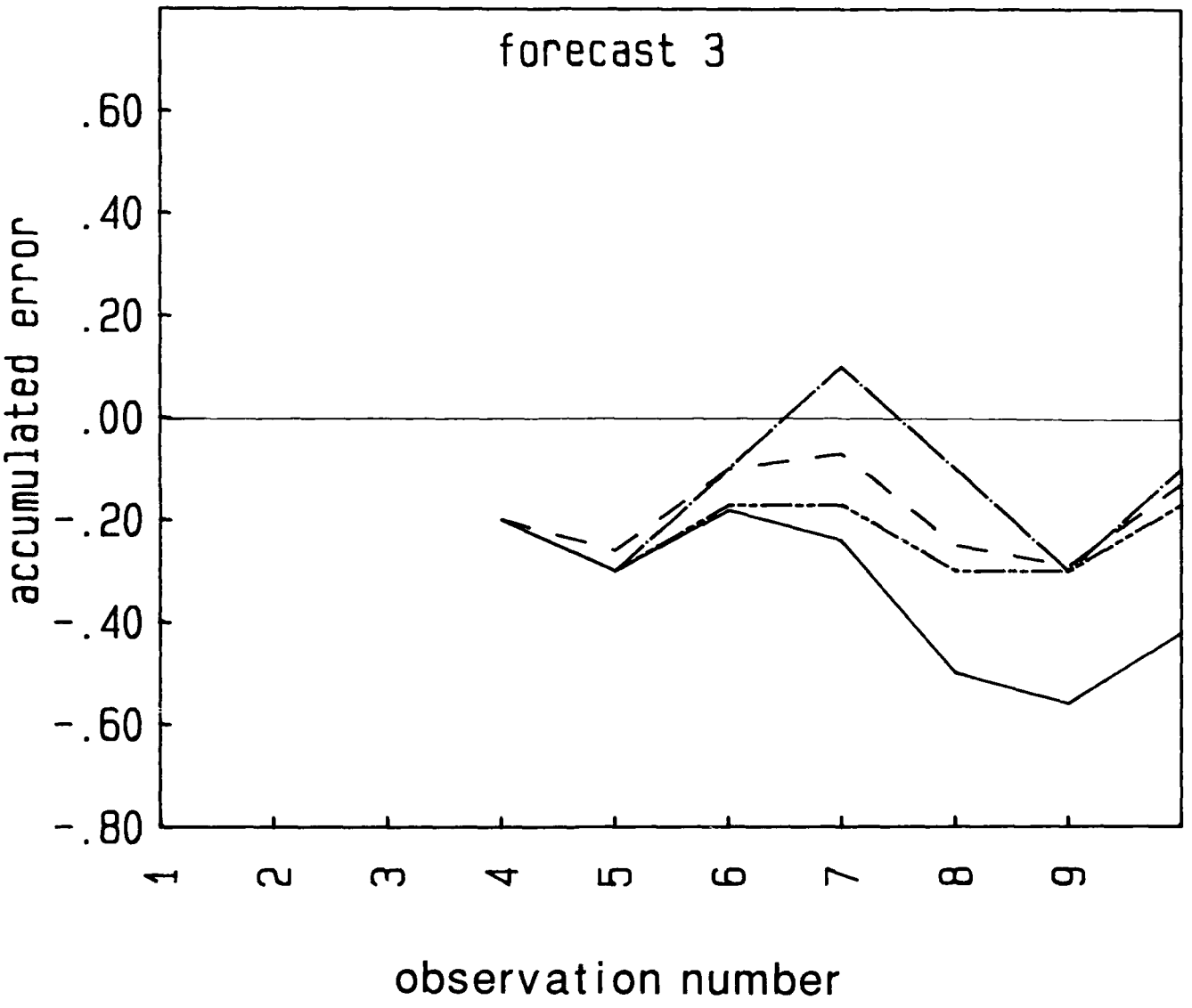

Figure 15c. Plots of accumulated forecast error corresponding to Figure 14 data

reflects the first three forecast errors being negative in value, or the generation of forecast underestimates.

The accumulated absolute forecast error provides a good basis for technique comparison and results are presented in Table 5. It should be noted that the range of input data values for the stationary test data is very small in comparison to the ranges encountered in the nonlinear test data and that forecast error should be expected to increase due both to the data range and degree of nonstationarity. In these comparisons, of interest are: (1) the response of individual routines to variation of weighting factors and data type; and (2) the relative performance between the techniques.

Results from use of the stationary routines on stationary input data indicate that all perform quite similarly, with slight preference given to the KALMAN method. The SIMEXP forecast error increases with increasing weight applied to the new observations, indicating that the historical mean is a better approximation to the population mean than the individual observations. This is certainly an accurate observation for this particular data set. When the forecast is derived primarily from the historical average the forecast relies upon a quantity which increasingly approximates the true population mean as the noise effects are filtered out. The trend of decreasing total error for a given routine from the one to three interval forecasts primarily reflects the smaller number of forecast errors contributing to this sum for the larger interval forecasts.

These stationary routines behave quite differently when linear and quadratic test data are introduced. The ADSIMEXP filter appears to be most accurate, reflecting the automatic adaptation of the weighting factor. The best SIMEXP filter performance is obtained with the largest weight applied to the new observations, reflecting better adjustment to the changing nature of new data and reduction of the influence of the historical average. The KALMAN filter, on the other hand, senses less variation among its forecast values than in the input data set and reduces the weight given to new input data while increasing the weight applied to the historical data. This causes the KALMAN method to spiral into a state of larger and larger forecast error. These initial results suggested that the ADSIMEXP and SIMEXP (weight = 0.7) routines are the preferred stationary techniques.

The nonstationary techniques may be similarly compared. With stationary input data the methods are all generally similar in response to the stationary routines except they incur slightly greater error due to the attempt to fit linear trends to the data. This is particularly true for the first one-, two-, and three-interval forecasts, as shown for the LINREG method in Figure 16. With no historical data to rely upon, LIN REG projects the trends observed from the first two data values into the future. The box filter approach of LINREG method offers slightly better performance. This results from averaging equally all available data from the periodic data source. With linearly varying input data, all linear trend methods naturally return zero error. The BWNQUAD routine, optimized for quadratic input data, employs a linear trend initialization scheme and suffers the greatest error among the nonstationary formulations. Even so, this error is still not significant when compared to the range of input data values, and all these techniques may be gauged satisfactory on stationary and linear test data.

Significant differences between routine performance are observed when quadratic test data are employed. The beneficial effect of preferential weighting of new input data (for example, increasing weight for HOLT2P and BWNQUAD, decreasing weight for BWN1PAD) is easily observed. These techniques have better absolute accuracy and show a greater relative increase in forecast accuracy than possible with the stationary SIMEXP routine with large preferential weight to new data. While

Table 5. Accumulated absolute forecast error obtained from application of generated test data with the forecasting techniques. Results for techniques allowing user-selection of weights are shown for the three weight values W = 0.3, 0.5, and 0.7.

DATA	SIMEXP			ADS IMEXP	SIMM OVAV	KAL MAN	LINM OVAV	LIN REGR	HOLT2P			BWNIPAD			BWNQUAD		
TYPE	WT .3	.5	.7						.3	.5	.7	.3	.5	.7	.3	.5	.7
								FIRST FORECAST INTERVAL									
MEAN(MEAN START)	1.3	1.5	1.7	1.7	1.4	1.3	2.2	1.8	2.6	2.0	2.3	2.2	2.0	2.1	1.7	1.8	1.7
MEAN(LOW START)	1.4	1.5	1.7	1.9	1.3	1.2	1.8	1.5	2.5	1.7	2.0	2.0	1.8	1.9	2.0	2.4	2.2
LINEAR	22.5	16.0	12.2	9.5	16.5	33.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0	1.4	0.5
QUADRATIC	217.0	164.0	130.0	100.0	173.0	356.0	46.0	72.0	110.0	56.0	31.0	29.2	48.0	87.2	44.0	9.1	1.5
								SECOND FORECAST INTERVAL									
MEAN(MEAN START)	1.1	1.3	1.4	1.5	1.1	1.0	2.5	2.1	3.1	2.2	2.9	2.8	2.5	2.7	2.7	3.1	4.1
MEAN(LOW START)	1.3	1.4	1.6	1.7	1.3	1.2	2.7	2.2	3.4	2.2	2.7	2.5	2.4	3.0	2.6	3.0	4.6
LINEAR	27.0	22.0	18.8	16.5	22.5	34.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.9	3.1	1.4
QUADRATIC	265.0	226.0	200.0	178.0	234.0	356.0	82.0	103.0	142.0	90.0	61.0	62.0	87.0	127.0	73.0	19.5	3.8
								THIRD FORECAST INTERVAL									
MEAN(MEAN START)	0.8	0.9	1.1	1.3	0.7	0.9	1.6	1.5	3.2	1.6	2.1	2.5	1.8	2.9	2.9	3.3	4.7
MEAN(LOW START)	.8	.9	1.0	1.3	0.6	0.8	2.2	2.2	3.9	2.2	2.7	3.1	2.5	3.6	3.1	3.2	6.0
LINEAR	30.2	26.0	23.4	21.5	26.5	34.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.4	5.2	2.6
QUADRATIC	295.0	268.0	249.0	230.0	274.0	350.0	117.0	129.0	167.0	122.0	93.0	96.0	121.0	157.0	99.0	32.0	7.0

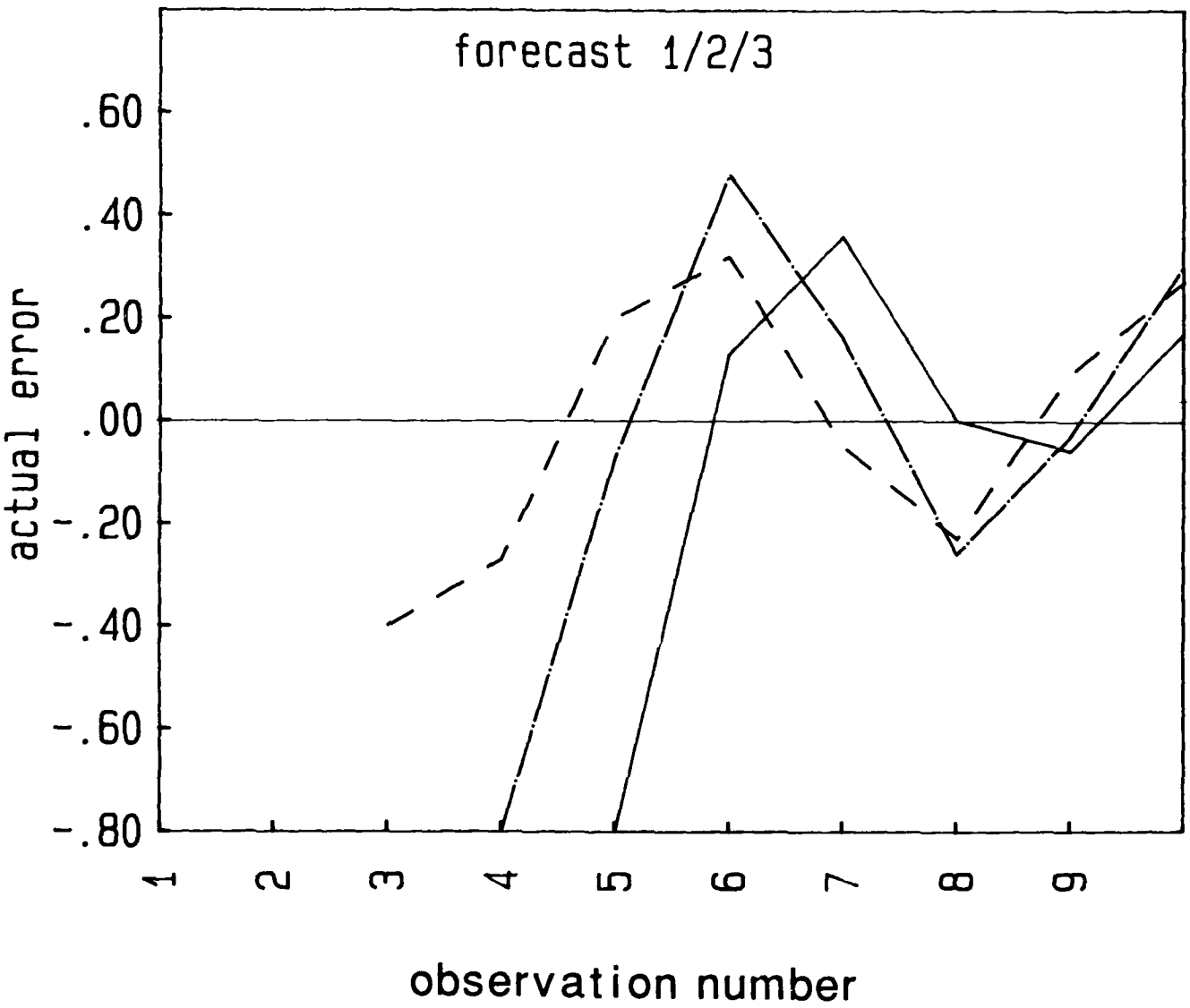

Figure 16. Plot of forecast error for LINREG method using stationary data (zero start) for one (dash), two (dash-dot), and three-interval (solid) forecasts

the SIMEXP method does adjust to a new "mean" it only projects into the future the current mean value.

Before wholehearted acceptance is given to any routine, other factors need to be considered. Consider the variation of actual forecast error with observation number for the BWNQUAD routine for the two cases of linear and quadratic input data as shown in Figures 17 and 18. The rapid decrease of forecast error magnitude shows that the method adjusts to the true trend if given a long enough historical data series and that most error occurs during the initial operation of the routine. This routine performs three recursive smoothing operations. For the first filter, three observations satisfy the data requirements where an average of the latest three observations is taken. However, the other two filter operations operate on insufficient data. Five observations are actually required to engage the true formulation to develop the first forecast. Thus this technique would appear most useful only for features that may be tracked for long periods of time and for which forecasts are not required quickly after first observation. This, however, does not address well the problem encountered here. A tradeoff must be made between deriving forecasts early without the use of an extensive data base and the desire to use the exact formulation.

Also, why not extend the usefulness of routines by employing "difference" data? This is particularly attractive for the stationary methods due to their computational simplicity. The results from use of such data are displayed in Table 6 and show the stationary methods can be effective for original data having well-behaved linear trends. Use of "differenced" data derived from the test stationary data results in slightly greater forecast errors since only noise serves as input to the routines. The relative ranking of the routines remains unchanged with the ADSIMEXP and SIMEXP methods being the most responsive of the stationary data methods.

Use of "differenced" data with the nonstationary routines results in unacceptable errors when applied to stationary data but results in improved forecasts for original nonstationary observations. The nonlinear methods all provide precise forecasts with the differenced data, although the BWNQUAD method again has slight error due to the differing assumptions employed in the initialization procedure and exact algorithm. There is little distinction between forecast capability of nonstationary data routines when applied to "difference" data.

These results provide some initial insight into selection of preferred routines. The generated test data suggested using the BWNQUAD (weight = 0.7) method for overall use if the features would be long lived. Alternatively, input data comprised of difference data could be used with a variety of methods, with preference given to LINREG, HOLT2P (weight = 0.5), BWN1PAD (weight = -0.5) and the LINMOVAV routines. If the data were known to be essentially stationary in overall character, then the stationary routines ADSIMEXP and SIMEXP (weight = 0.7) would appear most appropriate.

5.3.2 REAL TEST DATA

Control data brought some insight into the mechanics and performance of the various routines. Another test data set, obtained from satellite and radar observations of Hurricane Gloria was also utilized. One data set, determined from GOES satellite IR imagery, consisted of the locations (X,Y) of the center of area (COA) of the high cirrus shield (here termed cold dome) associated with the hurricane. A second data set consisted of COA positions of hurricane rainbands as determined from weather radar precipitation intensity contours. The results were somewhat unexpected. They

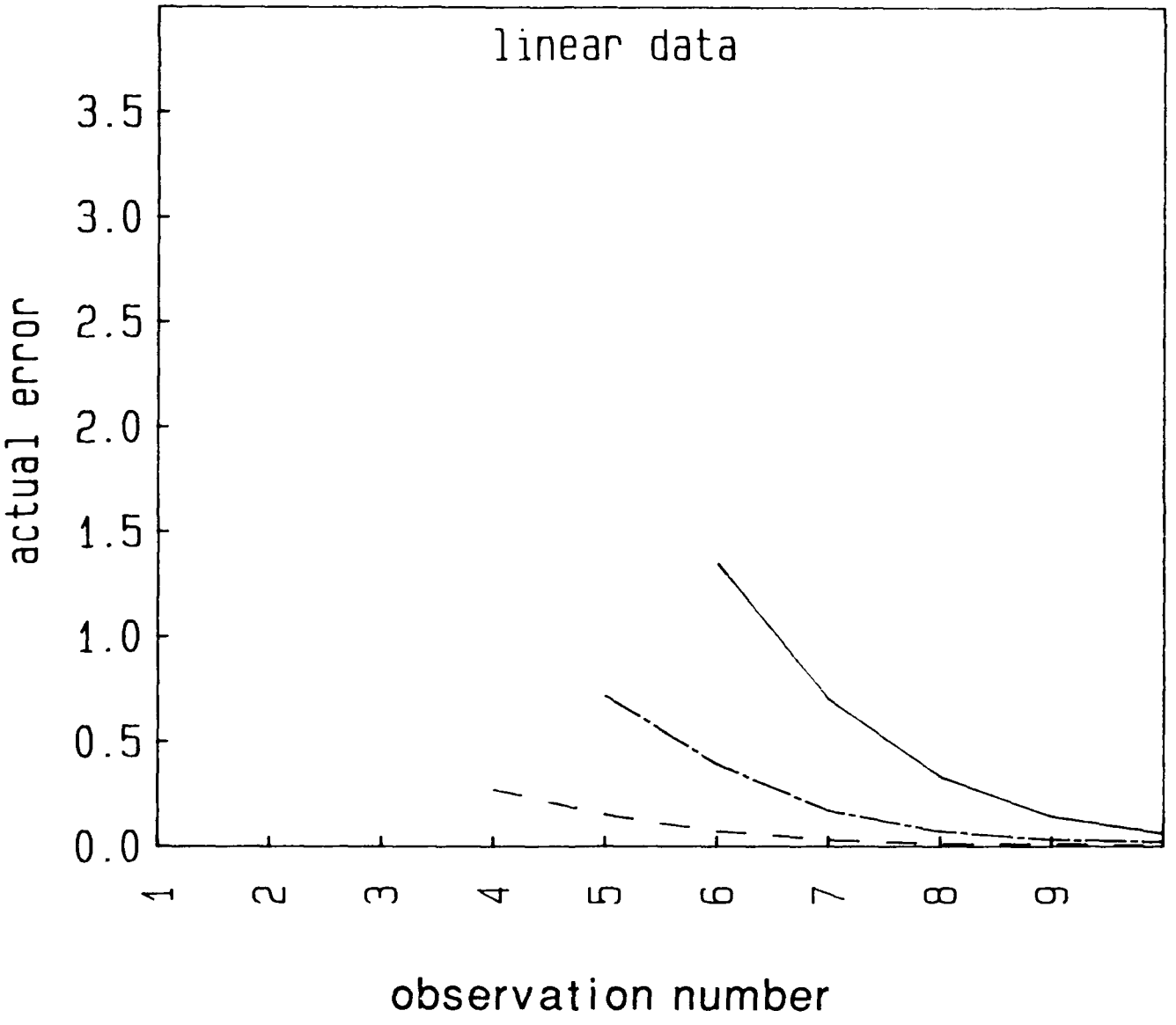

Figure 17. Plot of forecast error for BWNQUAD routine using linear data for one (dash), two (dash-dot), and three-interval (solid) forecasts

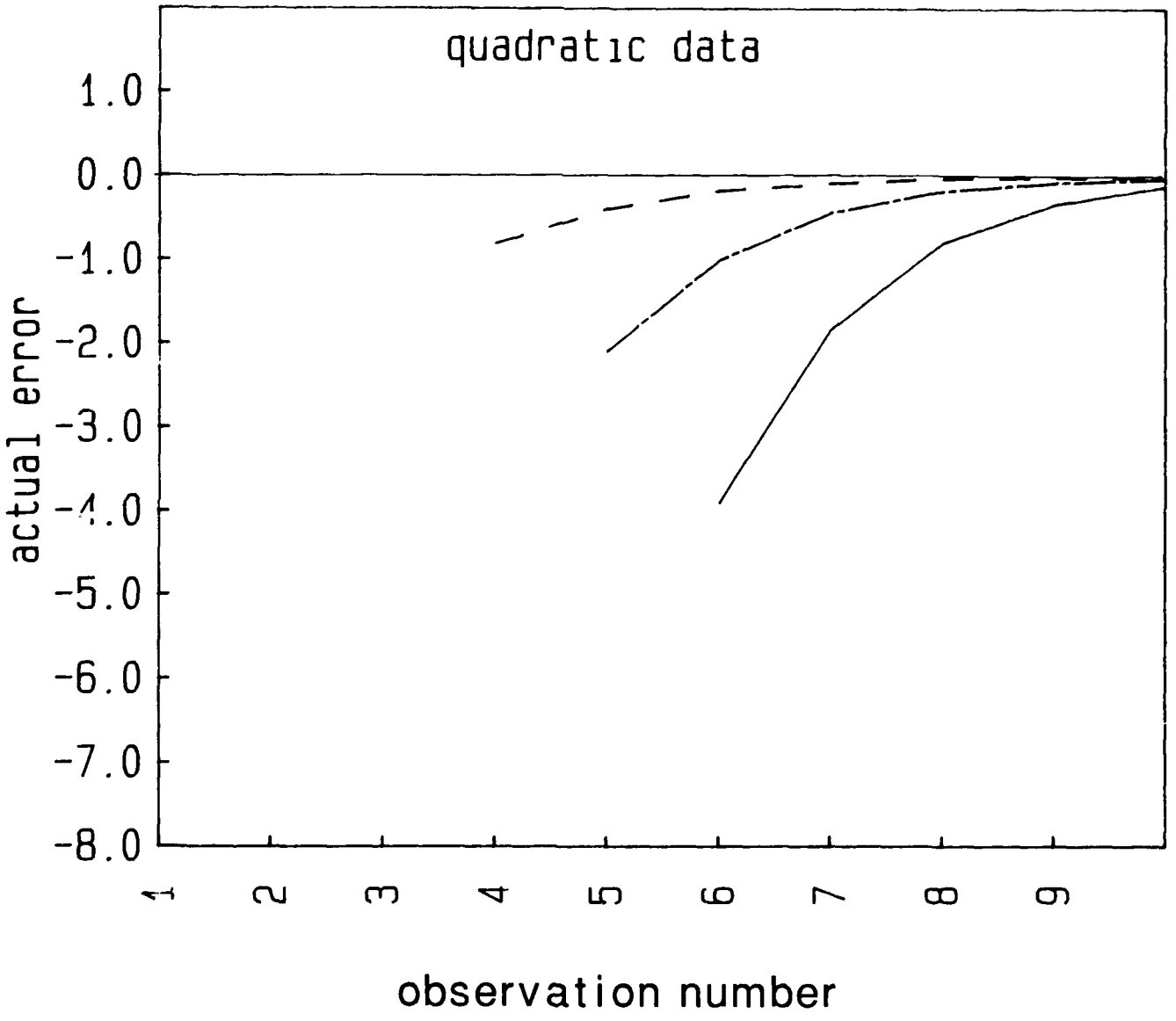

Figure 18. Plot of forecast error for BWNQUAD routine using quadratic data for one (dash), two (dash-dot), and three-interval (solid) forecasts

Table 6. Accumulated absolute forecast error obtained from application of generated test difference data with the forecasting techniques. Results for techniques allowing user-selection of weights are shown for the three weight values W = 0.3, 0.5, and 0.7.

DATA	SIMEXP			ADS IMEXP	SIMM OVAV	KAL MAN	LINM OVAV	LIN REGR	HOLT2P			BWNIPAD			BWNQUAD		
TYPE	WT .3	.5	.7						.3	.5	.7	.3	.5	.7	.3	.5	.7
						FIRST FORECAST INTERVAL											
MEAN(MEAN START)	2.0	2.0	2.0	1.8	2.2	2.1	3.3	2.3	4.8	2.6	2.4	2.3	2.4	3.7	1.6	2.0	3.0
MEAN(LOW START)	1.8	1.9	1.8	1.6	2.0	1.6	2.6	2.2	1.9	2.1	2.2	2.3	1.9	1.9	1.6	2.1	3.0
LINEAR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.5	0.4
QUADRATIC	38.7	28.0	21.6	17.0	29.0	53.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.2	2.8	0.7
						SECOND FORECAST INTERVAL											
MEAN(MEAN START)	2.6	2.9	3.5	4.0	2.7	2.6	6.7	4.7	10.3	5.8	6.5	6.7	6.0	8.6	3.2	5.6	7.5
MEAN(LOW START)	2.5	2.6	3.2	4.0	2.7	3.2	4.0	4.0	2.2	3.1	4.9	5.5	3.9	2.5	2.5	5.3	8.3
LINEAR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	1.4	1.3
QUADRATIC	79.0	62.0	52.0	44.0	64.0	97.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.1	8.5	2.5
						THIRD FORECAST INTERVAL											
MEAN(MEAN START)	2.1	2.7	3.6	4.2	1.8	2.4	8.8	6.8	15.1	8.2	9.3	9.5	8.1	12.8	3.4	7.3	11.7
MEAN(LOW START)	2.7	3.2	3.9	4.8	2.4	4.0	4.8	5.0	1.9	3.8	6.9	7.1	4.9	2.8	3.3	7.7	11.8
LINEAR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.4	2.3	3.1
QUADRATIC	115.0	96.0	84.0	75.0	99.0	130.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	10.2	17.1	5.7

illuminated some deficiencies in prospective forecasting techniques and point to the necessity for testing potential techniques in a variety of data environments. The IR cold dome COA locations along the easterly (x) and northerly (y) directions are shown in Figures 19a and b. Figure 19a shows the locations for the inner contour (labeled I in Figure 20). The larger contour O was also utilized, but because the results from these two data sets are quite similar, only results for contour I will be discussed in detail.

The forecast error from use of stationary routines with velocity (differences between COA observations) data are shown in Figures 21a-c and 22a-c. These plots represent the best performances of the individual routines where user-selected weight flexibility was available. The accumulated absolute forecast errors are shown in Tables 7 and 8. Overall, greater error is noted with use of x rather than y position data. Figures 19a and b show that, except for the midpoint deviation, the latter portion of the x-data series deviates from the initial trend while the y-data series returns to values suggested by the initial trend. These data series may be loosely termed quadratic in x and linear in y. Utilization of position differences as input reduces the data to nearly linear and constant behavior, respectively. Some observations previously cited were again noted. The ADSIMEXP routine overreacts to changes in the data trend and exhibits the greatest range of error, while the KALMAN response is sluggish.

Of greater importance, however, are the differences now noted with use of real, somewhat erratic, data. As shown in Figures 21 and 22 the SIMEXP routine performs best with use of a low weight (weight = 0.3) value, indicating the preferential weighting of the historical data trend over new observations. All the stationary routines exhibited greater error with use of x position data. The KALMAN filter, prone to large error with generated test data, behaves very respectably here. The sluggish response of the KALMAN routine results in greater x-component, and smaller y-component error than the other stationary data routines (except ADSIMEXP). This is explained by observation of the data and the operation of the routine. The KALMAN routine employs a weighted sum of the most recent observation and forecast, with the greatest weight applied to the most stable term (that is, exhibits the smaller variance). Sensing a sudden variation in observation (data input position 4), reduced weight is applied to new observations and emphasis is placed on past forecasts in developing the new forecast. Naturally, this results in maintenance of past forecast trends.

The responses of nonstationary routines to actual position data (not "difference" data) are shown in Figures 23a-c and 24a-c. The relative performance of these routines also generally differs from that observed with use of generated test data. The BWNQUAD routine strongly overreacts to the erratic variations from the long-term trend in the input data. No preferential weight value is found to bring its response in line with the other nonstationary techniques. In fact, the BWNQUAD results are not shown for the two and three interval forecasts (y position data) in Figures 24b and c because the values lie outside the plot range. The BWNQUAD routine appears to be of little use where the data exhibits sudden and significant changes from any general long-term trend. The LINMOVAV routine also responds poorly and has a sluggish response remininscent of the KALMAN method. The two techniques that consistently perform best are the BWN1PAD (weight = -0.5) and HOLT2P (weight = 0.5) routines.

The results for the LINREG method for all forecast intervals are displayed in Figure 25. Comparison with other nonstationary data techniques (Figures 23 and 24) shows that this method performs similarly to the BWN1PAD routine with use of the IR inner contour position data, but

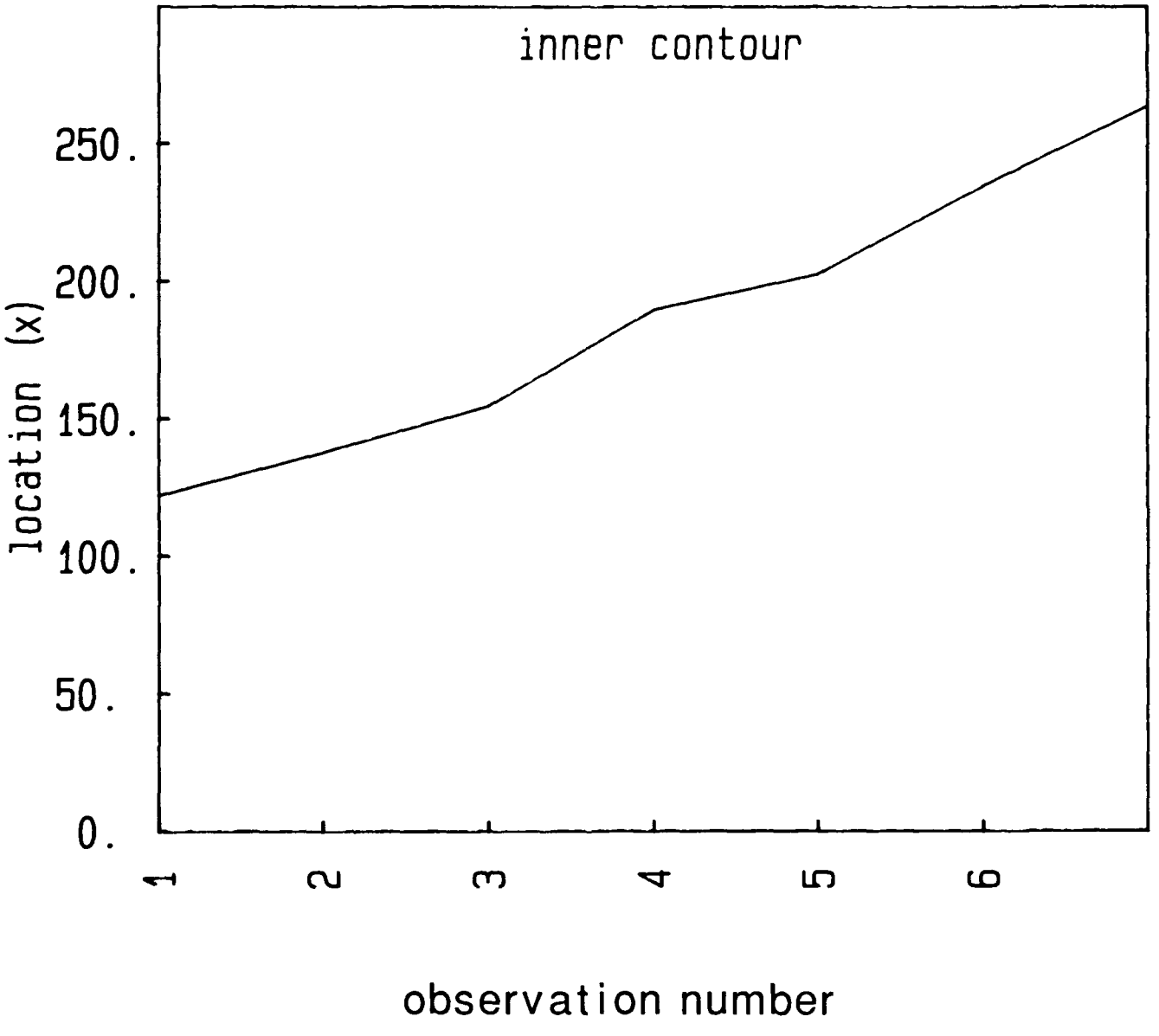

Figure 19a. Plots of center of area (COA) image pixel locations for X (east) direction for inner high cirrus area from IR satellite imagery of Hurricane Gloria

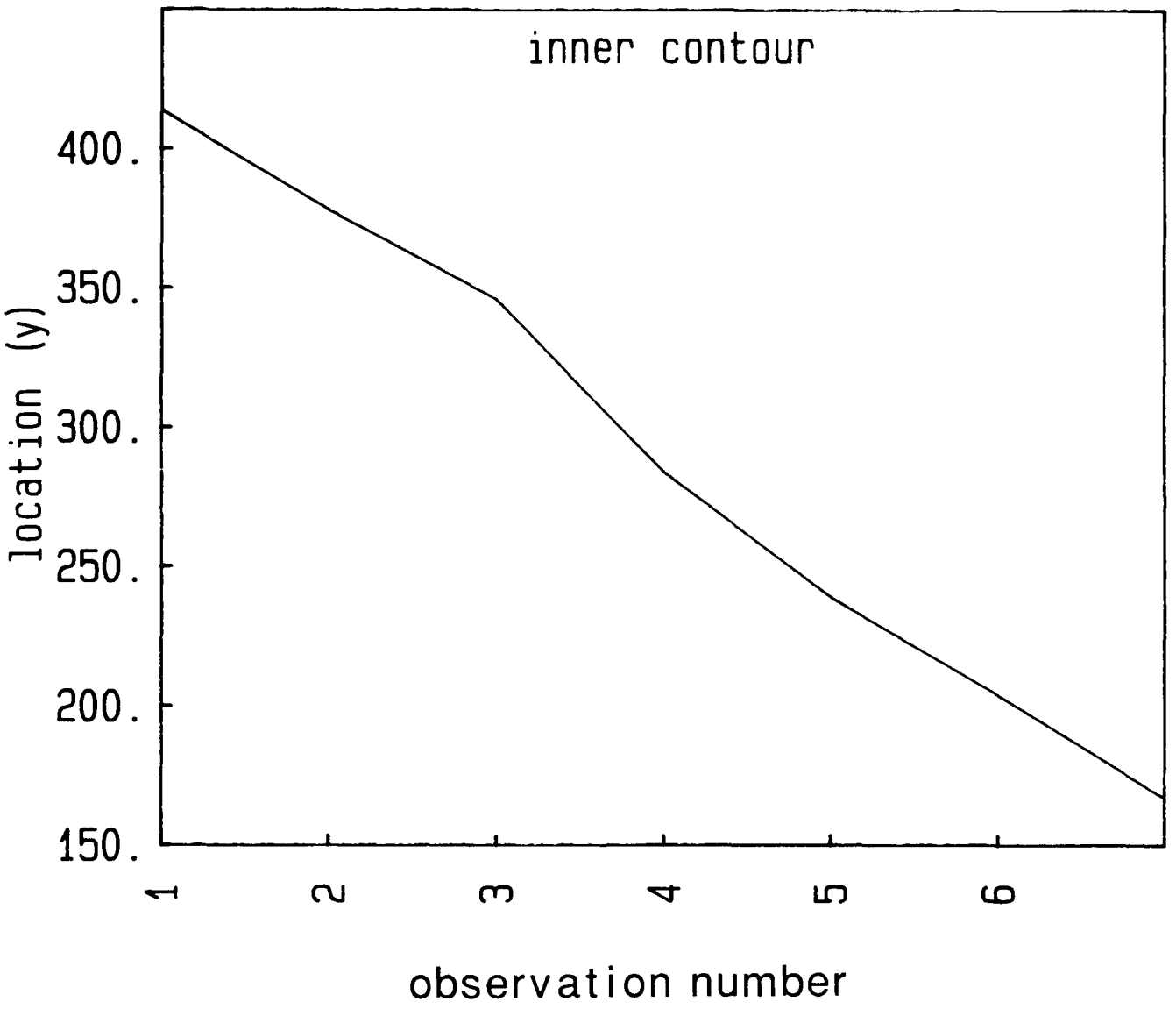

Figure 19b. Plots of center of area (COA) image pixel locations for Y (north) direction for inner high cirrus area from IR satellite imagery of Hurricane Gloria

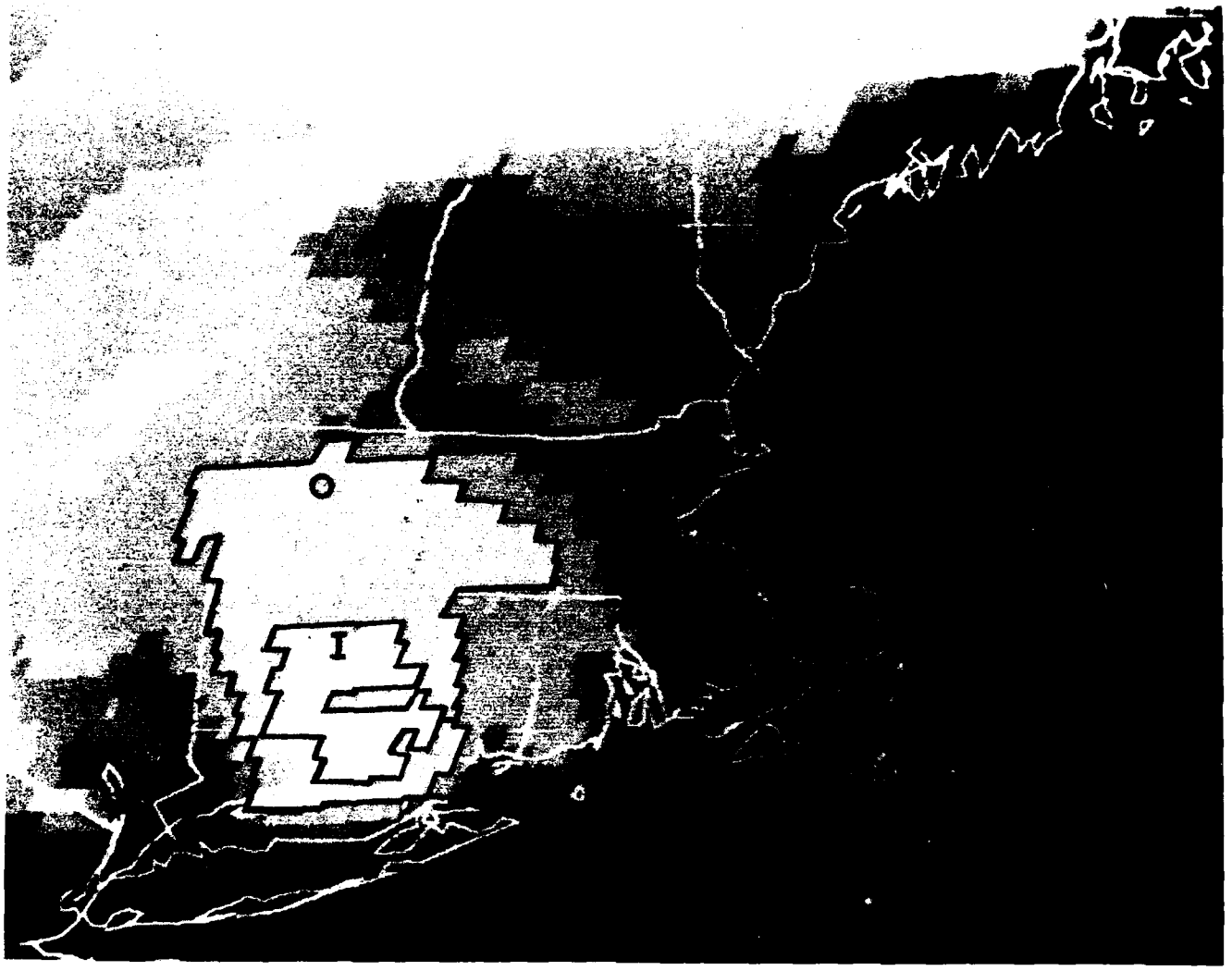

Figure 20. Plot showing the two IR cold dome contours tracked using GOES satellite imagery of Hurricane Gloria: inner (I), and outer (O) contours

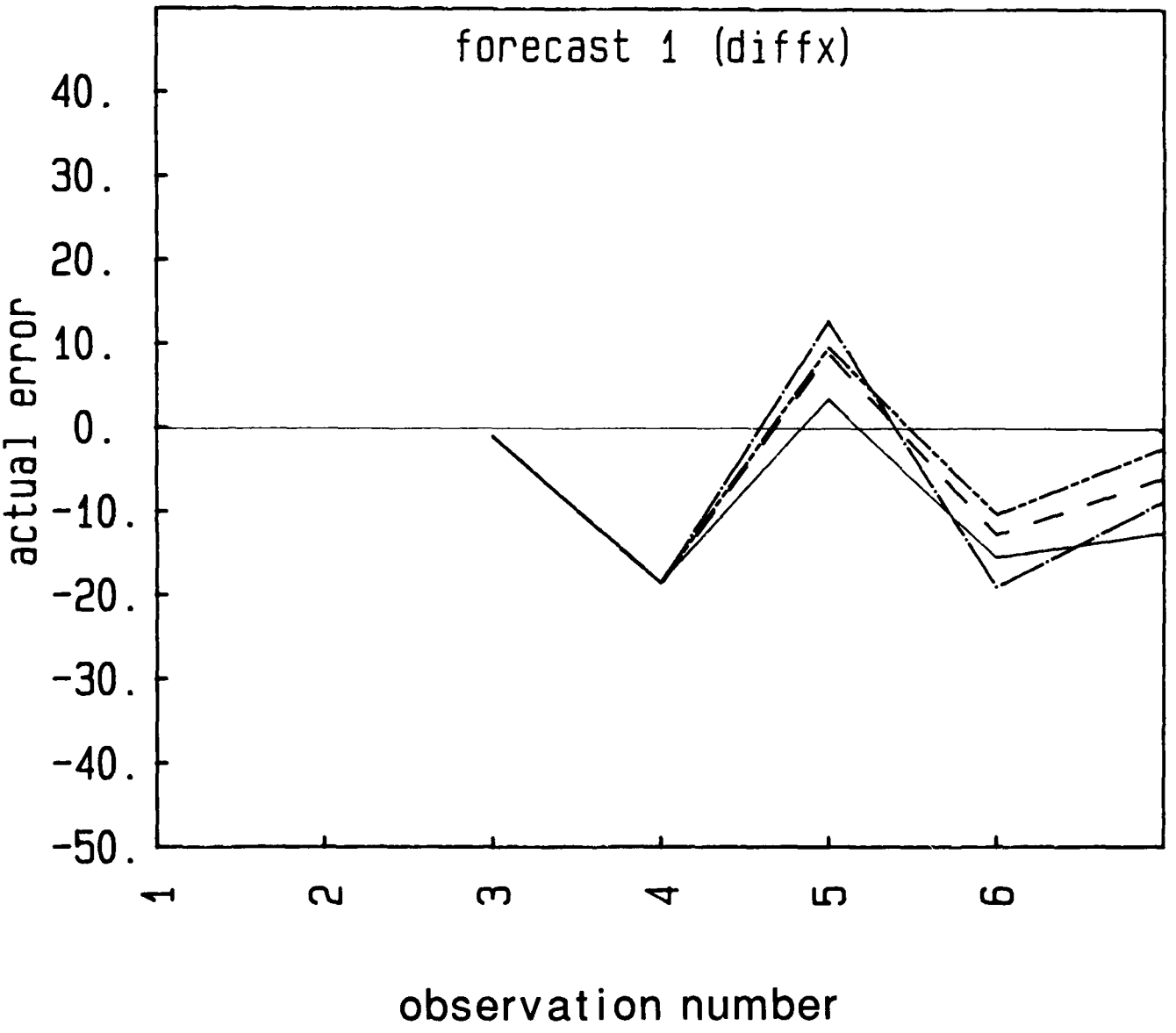

Figure 21a. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 19 for one-interval forecasts

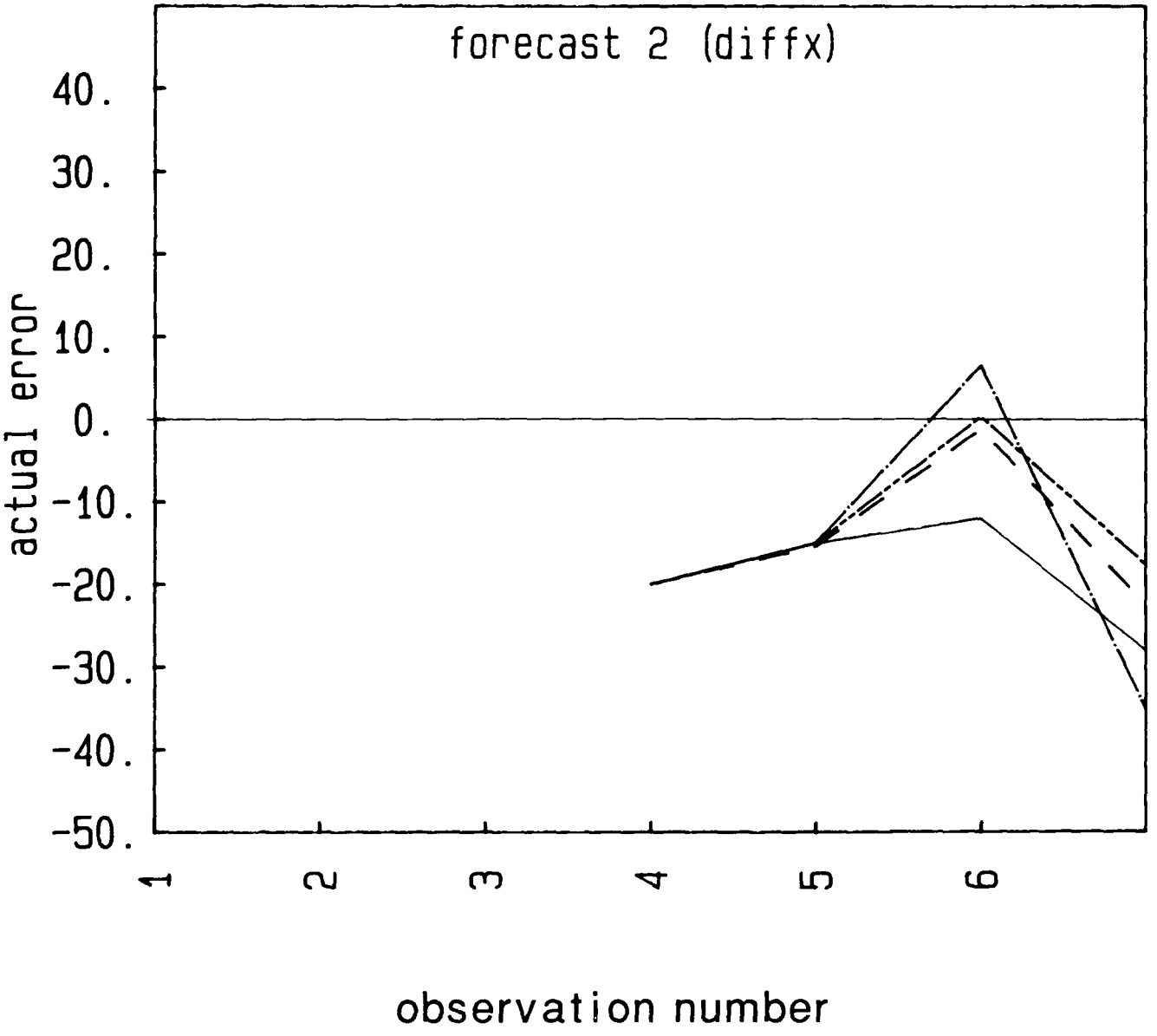

Figure 21b. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 19 for two-interval forecasts

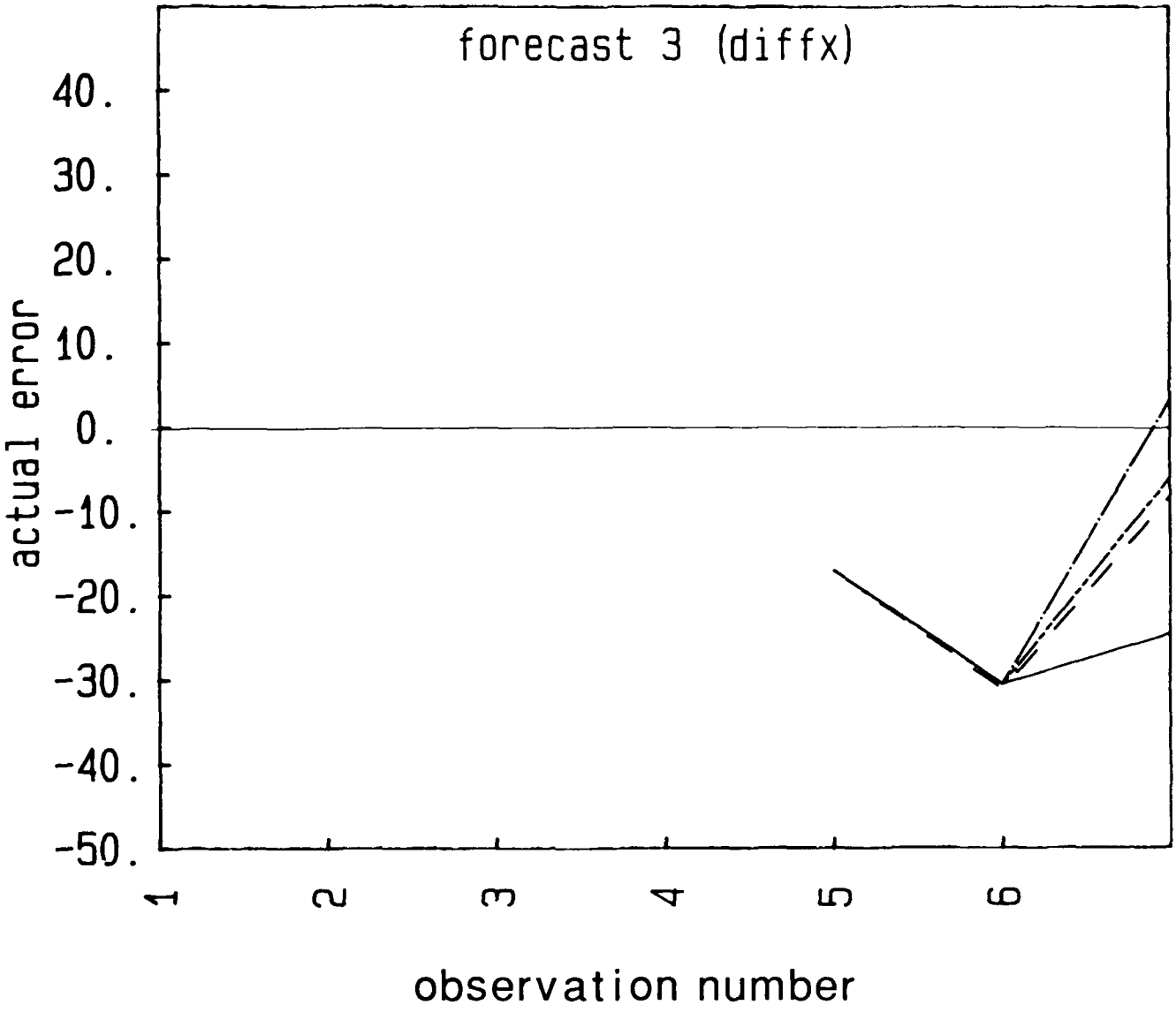

Figure 21c. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 19 for three-interval forecasts

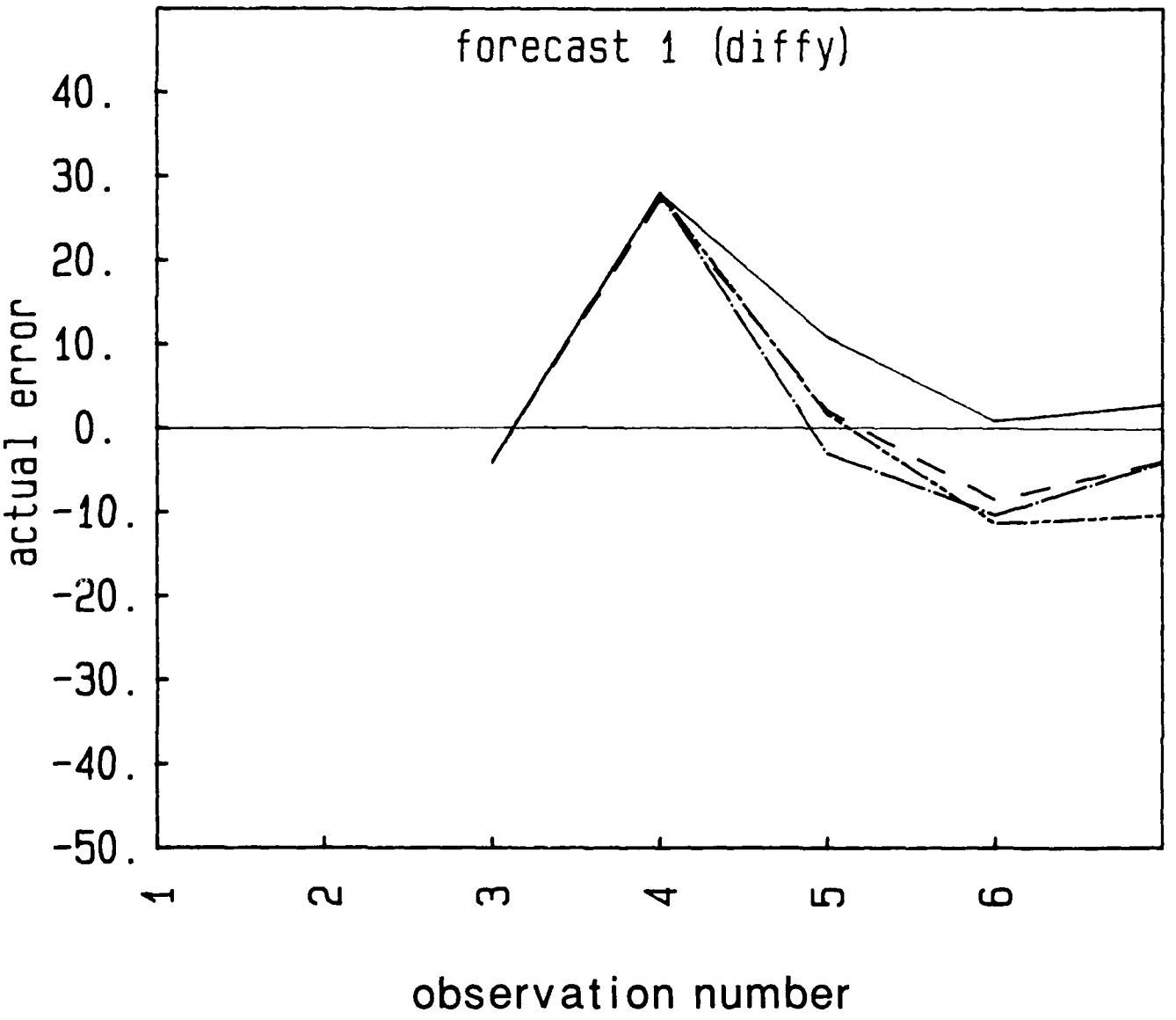

Figure 22a. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 19 for one-interval forecasts

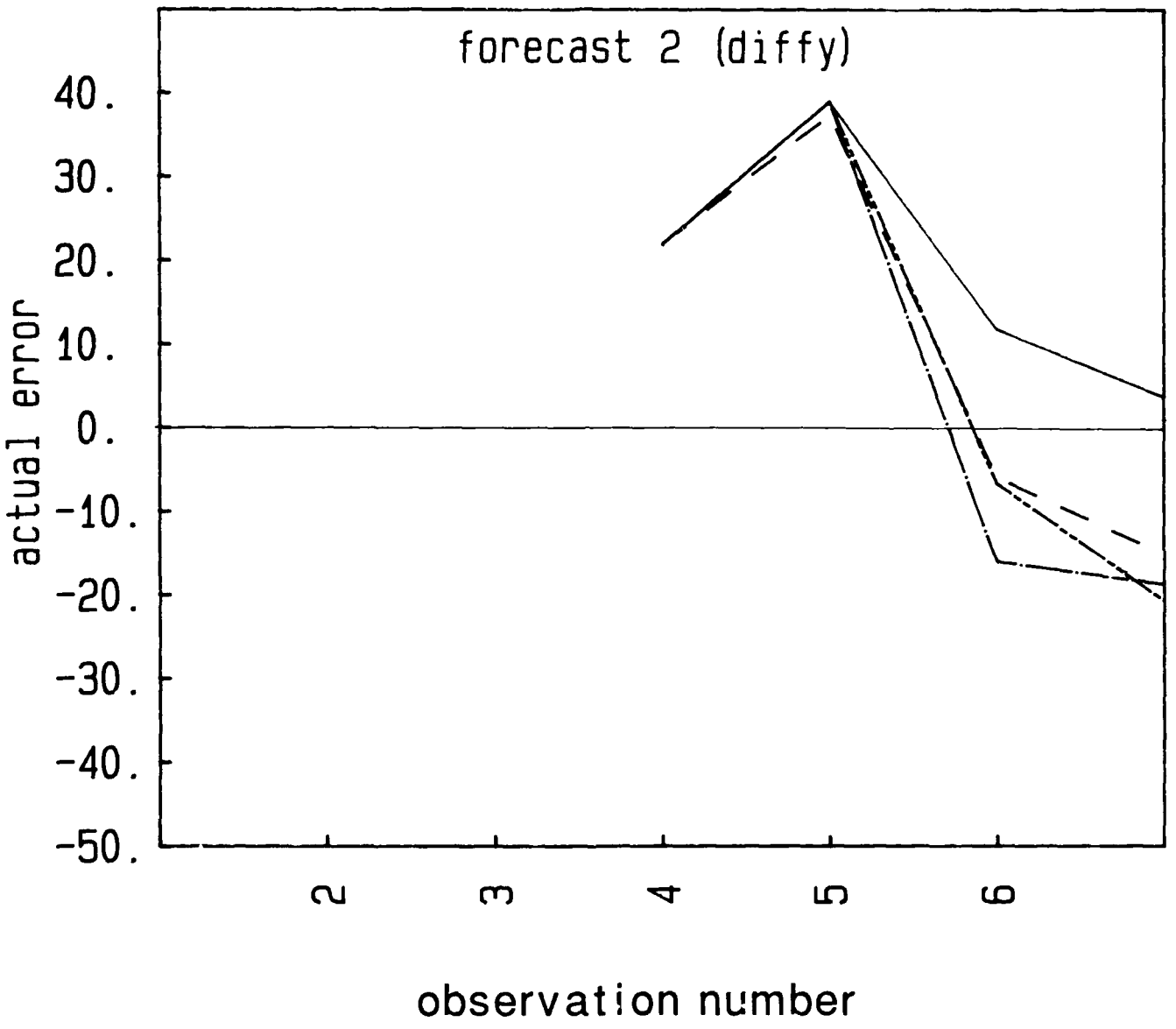

Figure 22b. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 19 for two-interval forecasts

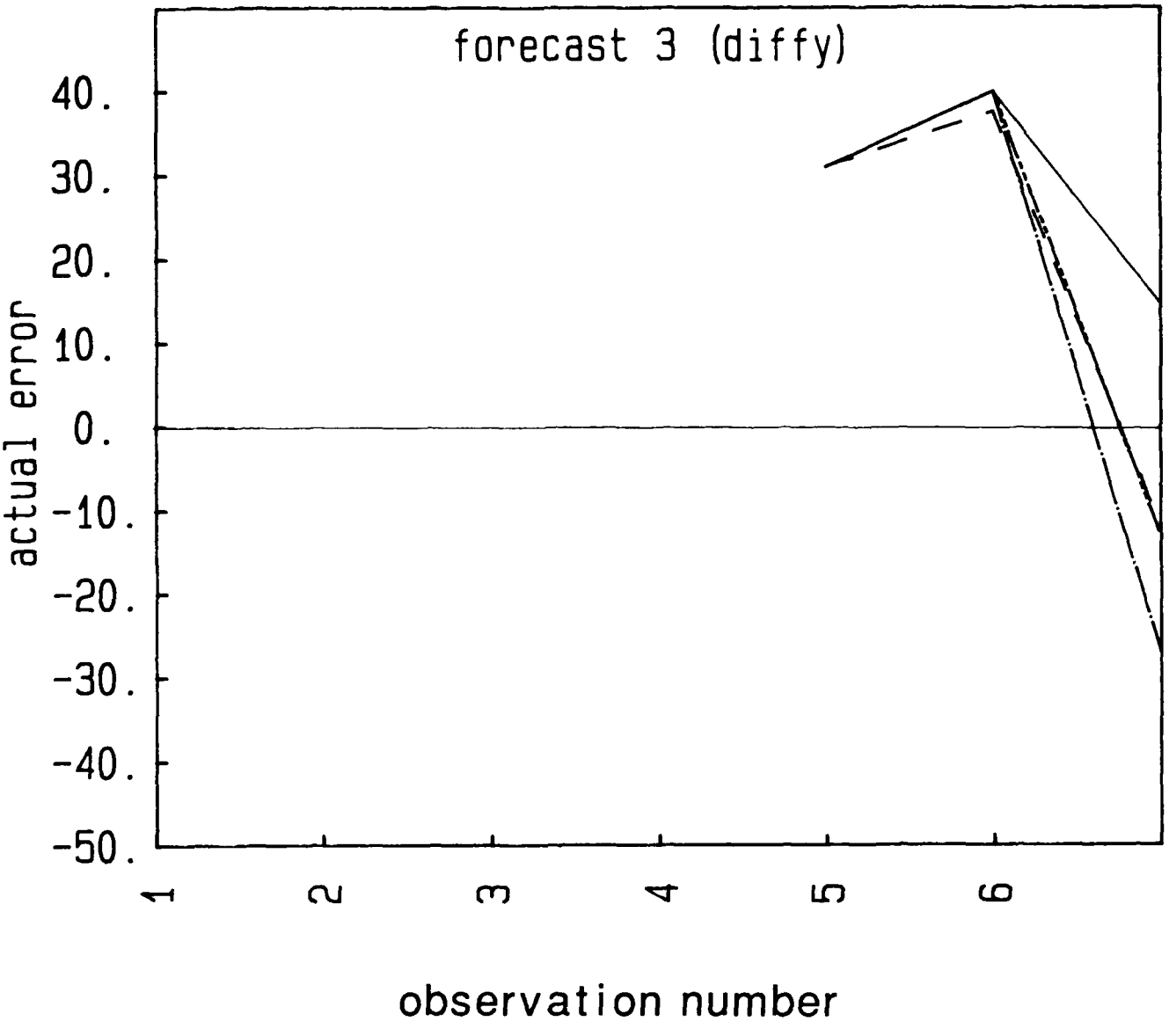

Figure 22c. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 19 for three-interval forecasts

Table 7. Accumulated absolute forecast error obtained from application of Center of Area data from inner IR contour of Figure 20 with the forecasting techniques. Stationary (nonstationary) techniques employ difference (original) data. Results for techniques allowing user-selection of weights are shown for the three weight values W = 0.3, 0.5, and 0.7.

DATA	SIMEXP			ADS IMEXP	SIMM OVAV	KAL MAN	LINM OVAV	LIN REGR	HOLT2P			BWNIPAD			BWNQUAD		
TYPE	WT .3	.5	.7						.3	.5	.7	.3	.5	.7	.3	.5	.7
	← POSITION DIFFERENCE DATA →						← ORIGINAL INPUT POSITION DATA →										
	FIRST FORECAST INTERVAL																
X	47.3	48.2	51.1	64.0	41.8	51.0	34.1	43.0	71.3	43.5	39.8	45.7	43.1	58.3	71.8	48.0	60.1
Y	45.8	50.3	55.4	55.3	55.3	46.5	68.9	56.4	67.3	56.3	60.4	49.7	53.5	53.0	520.2	233.3	171.5
	SECOND FORECAST INTERVAL																
X	59.1	63.8	73.7	76.5	53.0	75.0	63.7	58.4	89.4	67.4	50.5	59.2	63.3	81.9	114.8	87.7	89.1
Y	80.5	98.0	112.3	99.4	88.3	76.4	81.0	76.5	84.7	77.5	92.3	96.3	75.0	74.3	737.4	339.3	475.2
	THIRD FORECAST INTERVAL																
X	56.4	50.8	61.4	50.8	53.5	72.0	69.6	71.0	83.6	69.2	50.4	48.8	64.3	79.2	123.0	98.1	77.12
Y	80.5	98.0	116.5	98.0	84.0	85.6	73.1	71.7	83.5	71.8	86.1	93.7	68.0	79.6	819.1	350.0	706.3

Table 8. Accumulated absolute forecast error obtained from application of Center of Area data from outer IR contour of Figure 20 with the forecasting techniques. Stationary (nonstationary) techniques employ difference (original) data. Results for techniques allowing user-selection of weights are shown for the three weight values W = 0.3, 0.5, and 0.7.

DATA	SIMEXP			ADS IMEXP	SIMM OVAV	KAL MAN	LINM OVAV	LIN REGR	HOLT2P			BWNIPAD			BWNQUAD		
TYPE	WT .3	.5	.7						.3	.5	.7	.3	.5	.7	.3	.5	.7
	← POSITION DIFFERENCE DATA →						← ORIGINAL INPUT POSITION DATA →										
	FIRST FORECAST INTERVAL																
X	47.8	49.2	56.3	72.0	45.3	52.8	45.8	61.2	99.8	55.6	38.4	45.2	45.3	79.2	88.8	39.8	61.8
Y	90.6	99.0	101.4	101.0	106.0	67.1	125.3	139.9	103.4	106.2	106.7	101.1	90.5	89.4	486.9	259.7	193.2
	SECOND FORECAST INTERVAL																
X	74.0	73.5	83.3	108.0	62.0	93.2	82.7	70.5	125.5	89.0	59.1	68.6	82.0	113.1	137.5	77.3	95.5
Y	160.1	181.0	193.9	203.0	175.0	117.4	149.0	241.9	127.5	139.5	185.0	182.4	149.5	126.1	678.9	408.7	544.9
	THIRD FORECAST INTERVAL																
X	79.4	78.5	90.6	113.0	73.0	95.4	97.6	118.5	120.4	97.5	73.6	76.6	90.5	113.7	147.1	96.8	103.9
Y	144.5	169.0	192.5	232.0	148.0	106.3	118.9	265.0	100.6	119.3	157.8	164.9	129.0	97.9	763.4	405.8	832.2

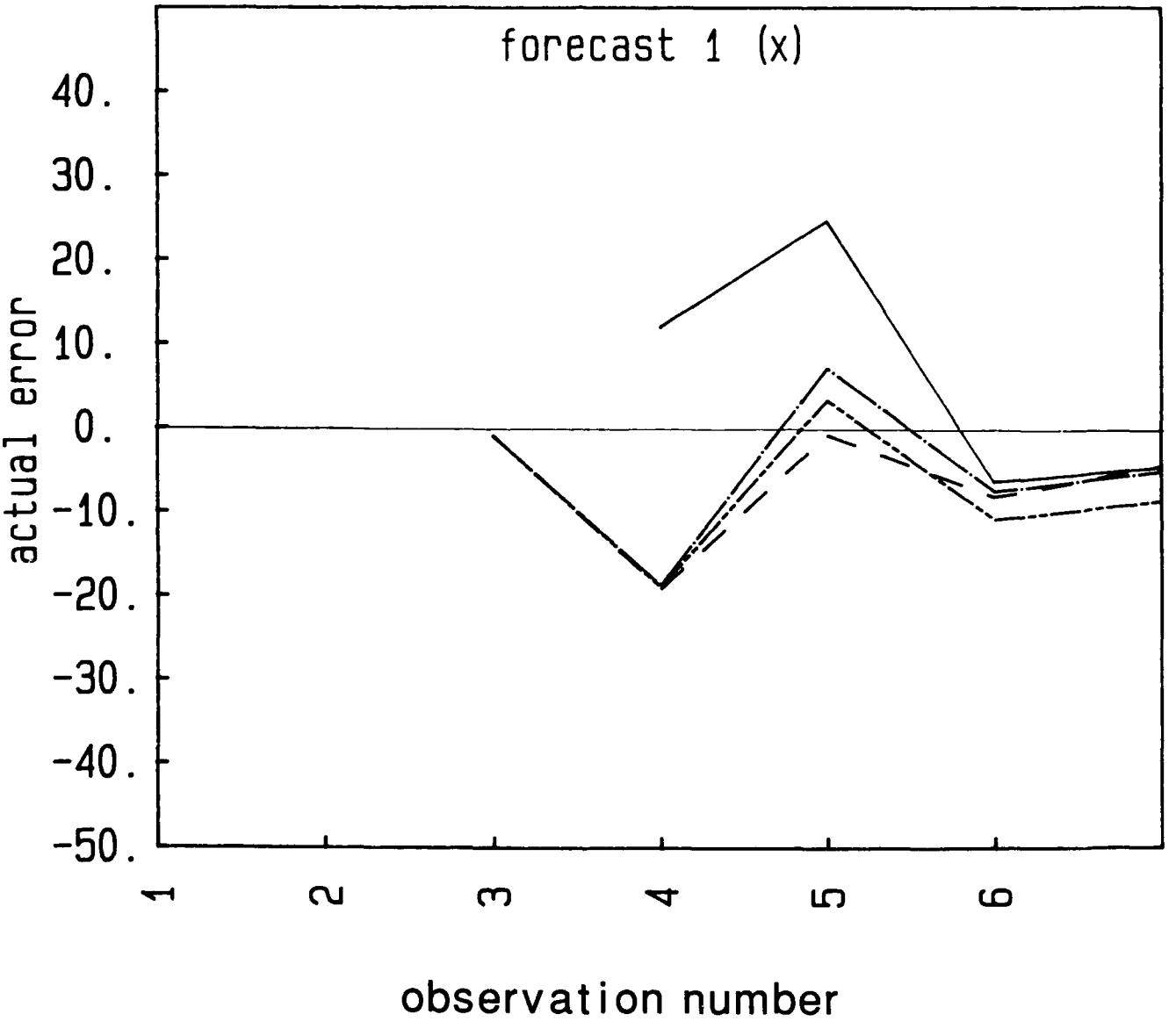

Figure 23a. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 19 for one-interval forecasts

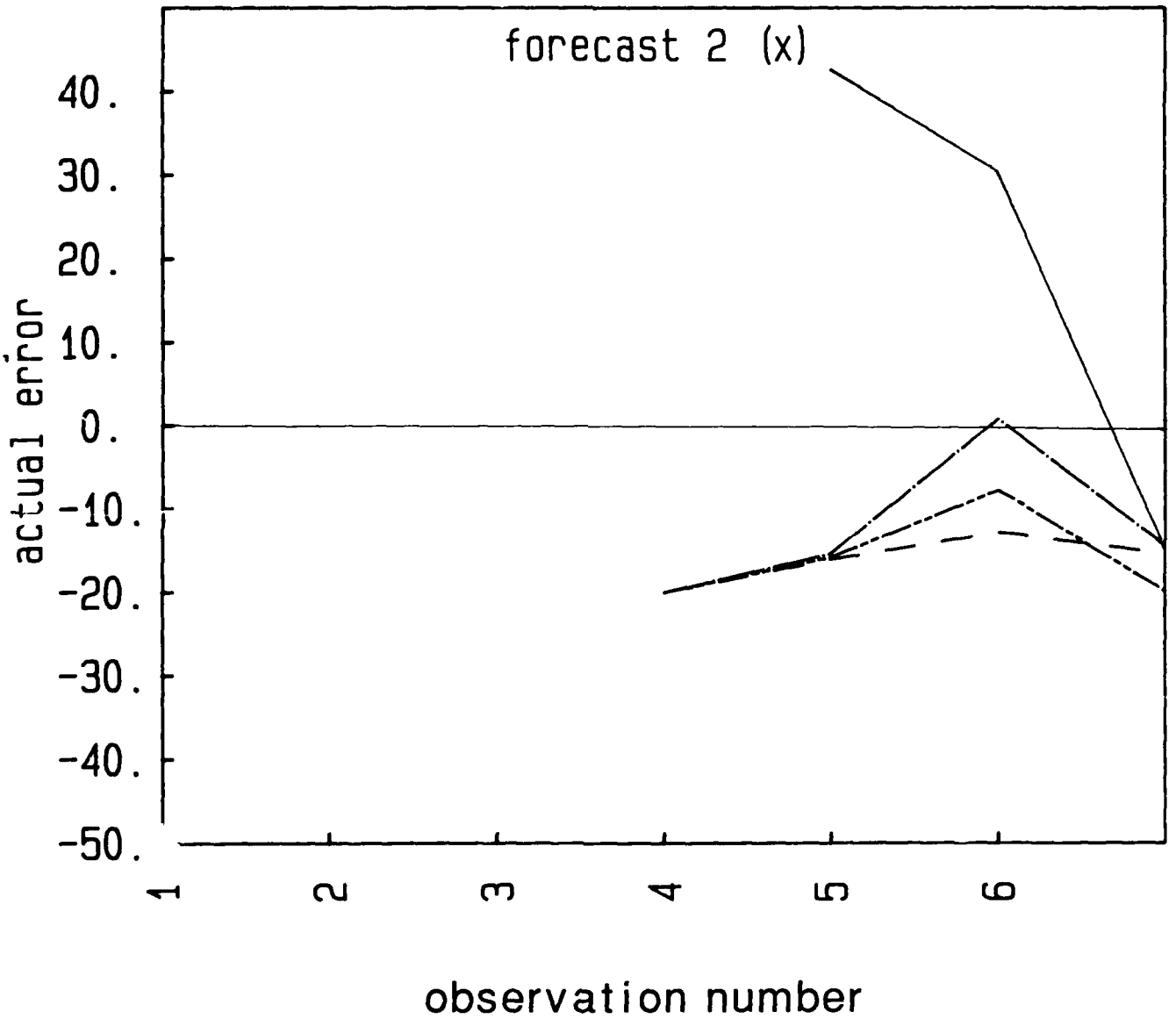

Figure 23b. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 19 for two-interval forecasts

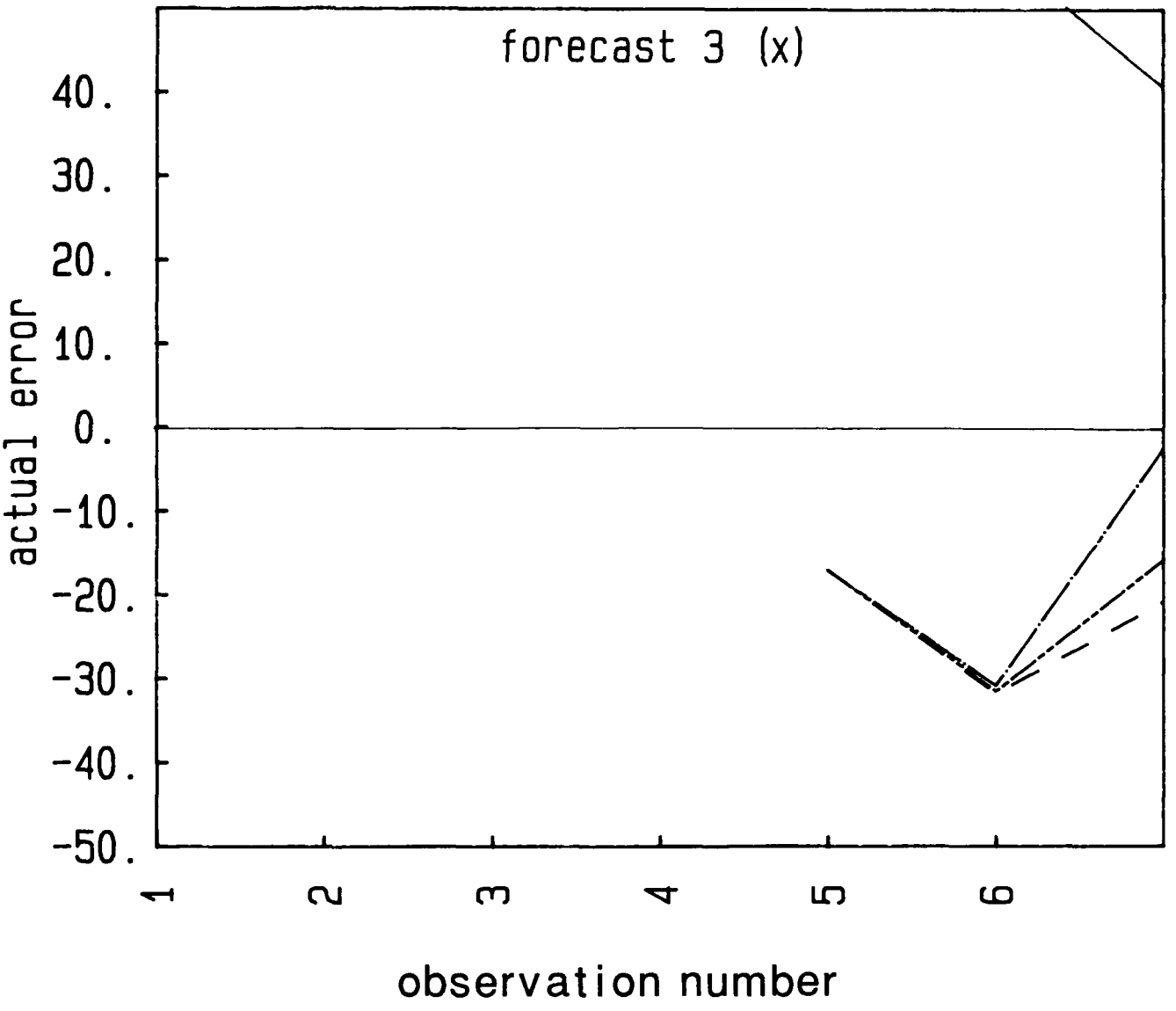

Figure 23c. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 19 for three-interval forecasts

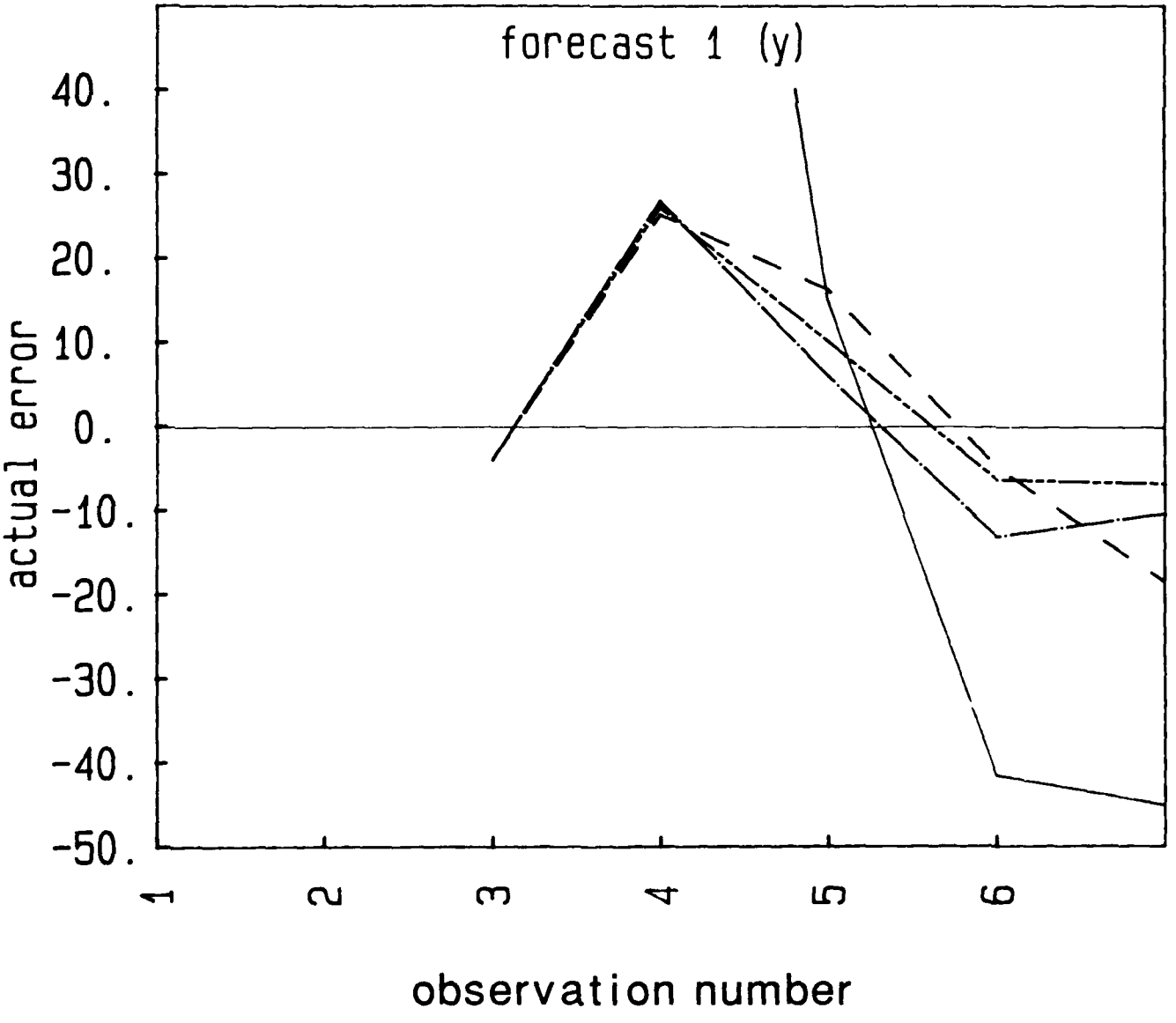

Figure 24a. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 19 for one-interval forecasts

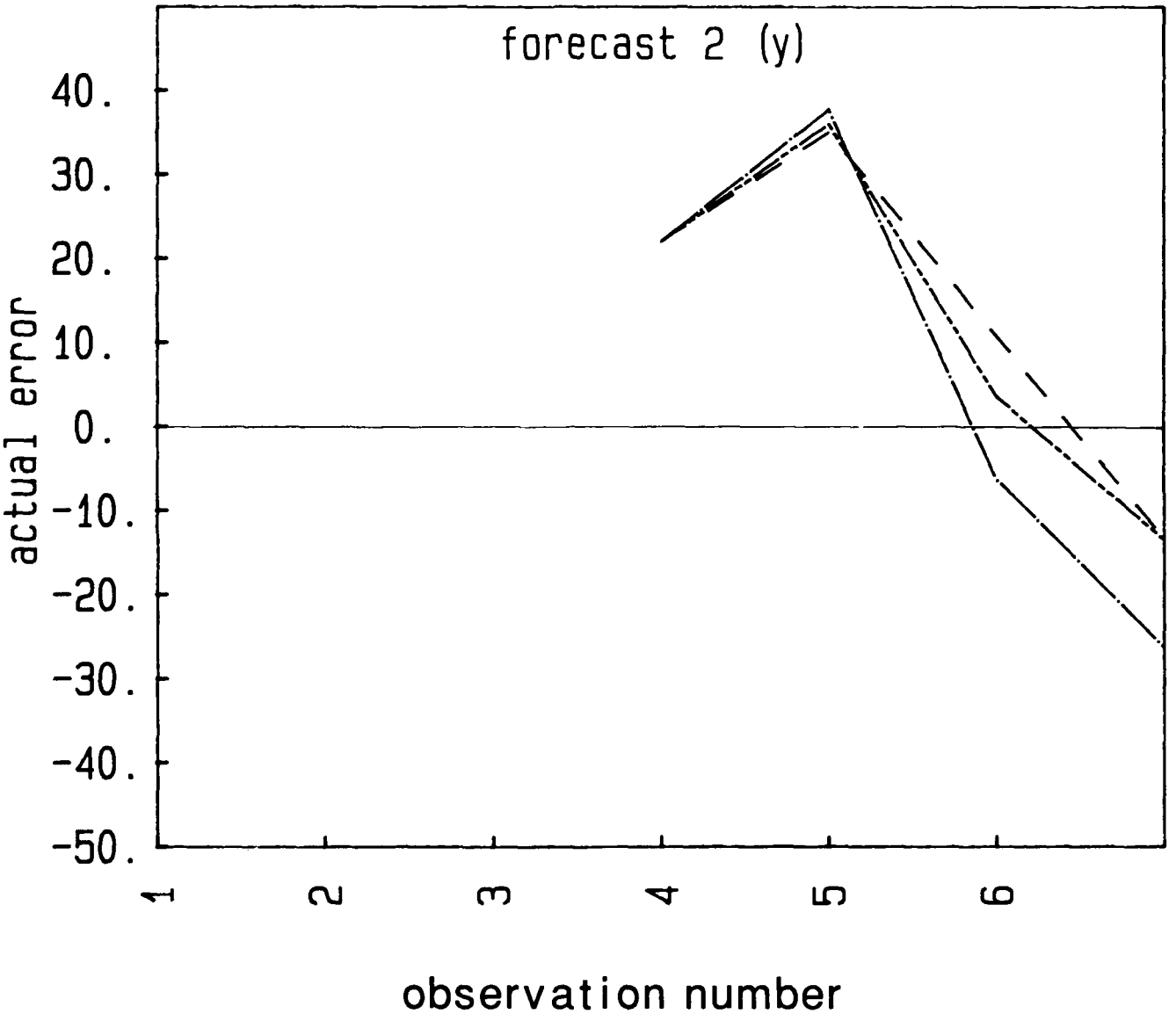

Figure 24b. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 19 for two-interval forecasts

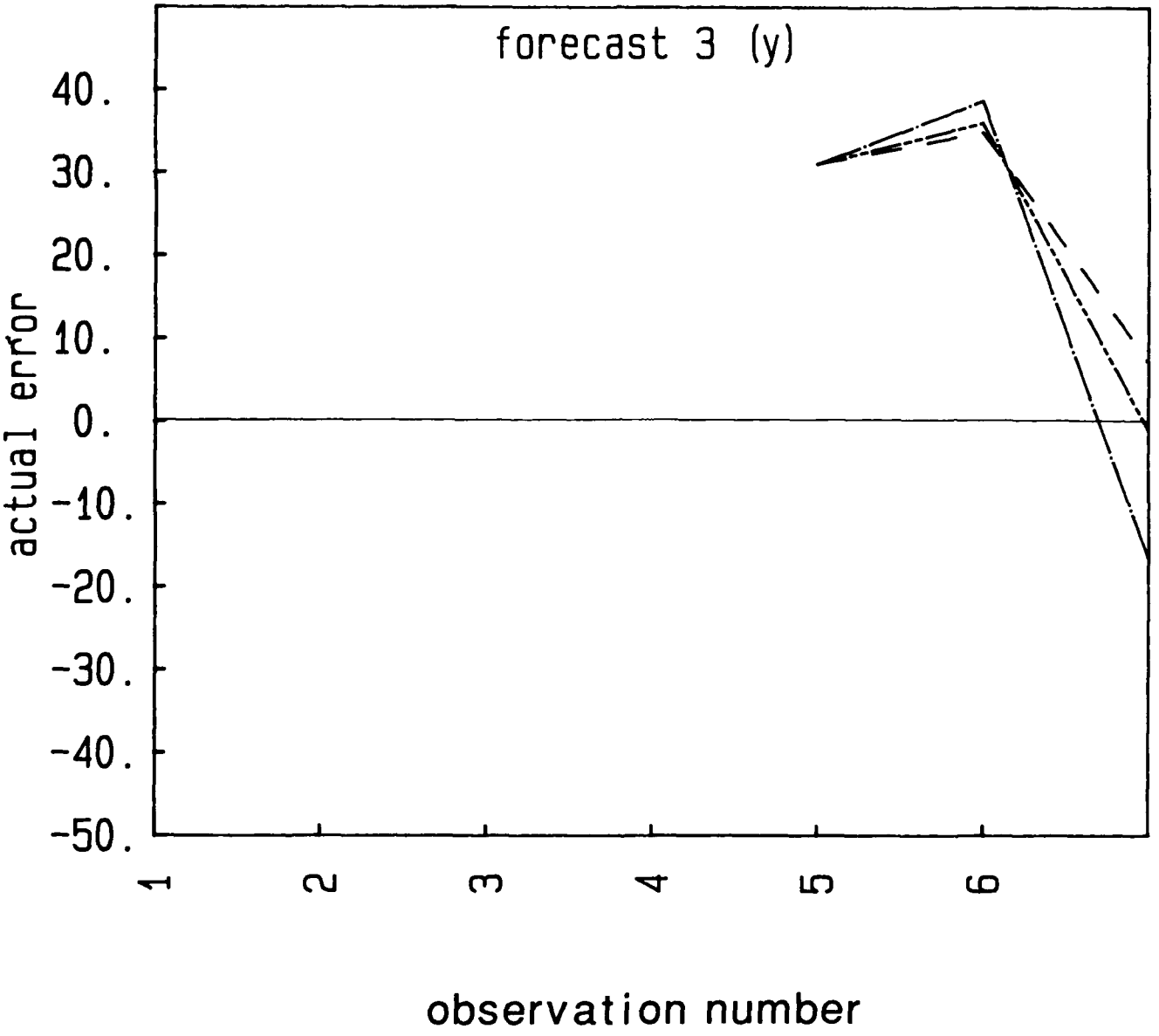

Figure 24c. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 19 for three-interval forecasts

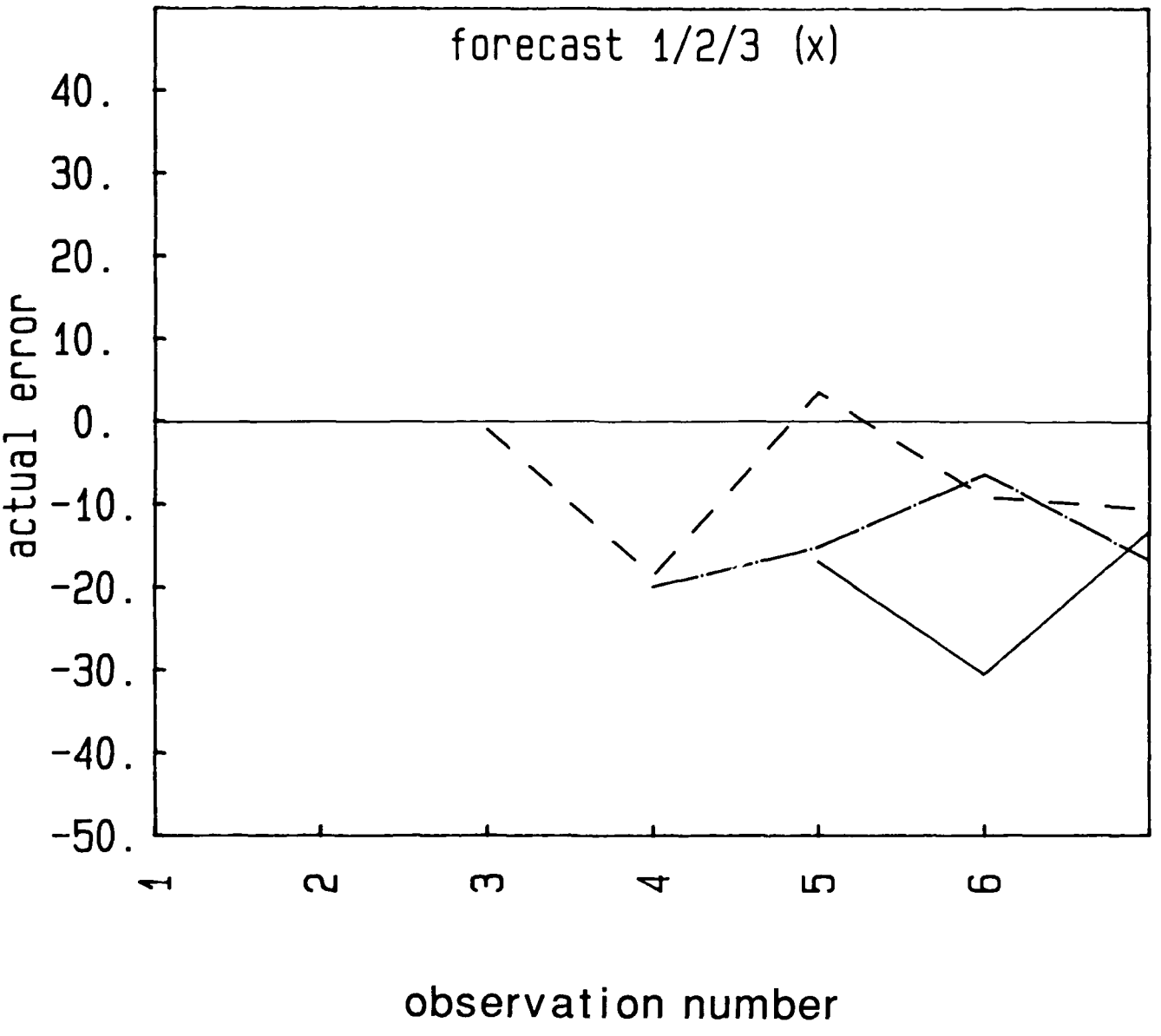

Figure 25a. Plot of forecast error for LINREG method using the position data of Figure 19 for one (dash), two (dash-dot), and three-interval (solid) forecasts: X position data

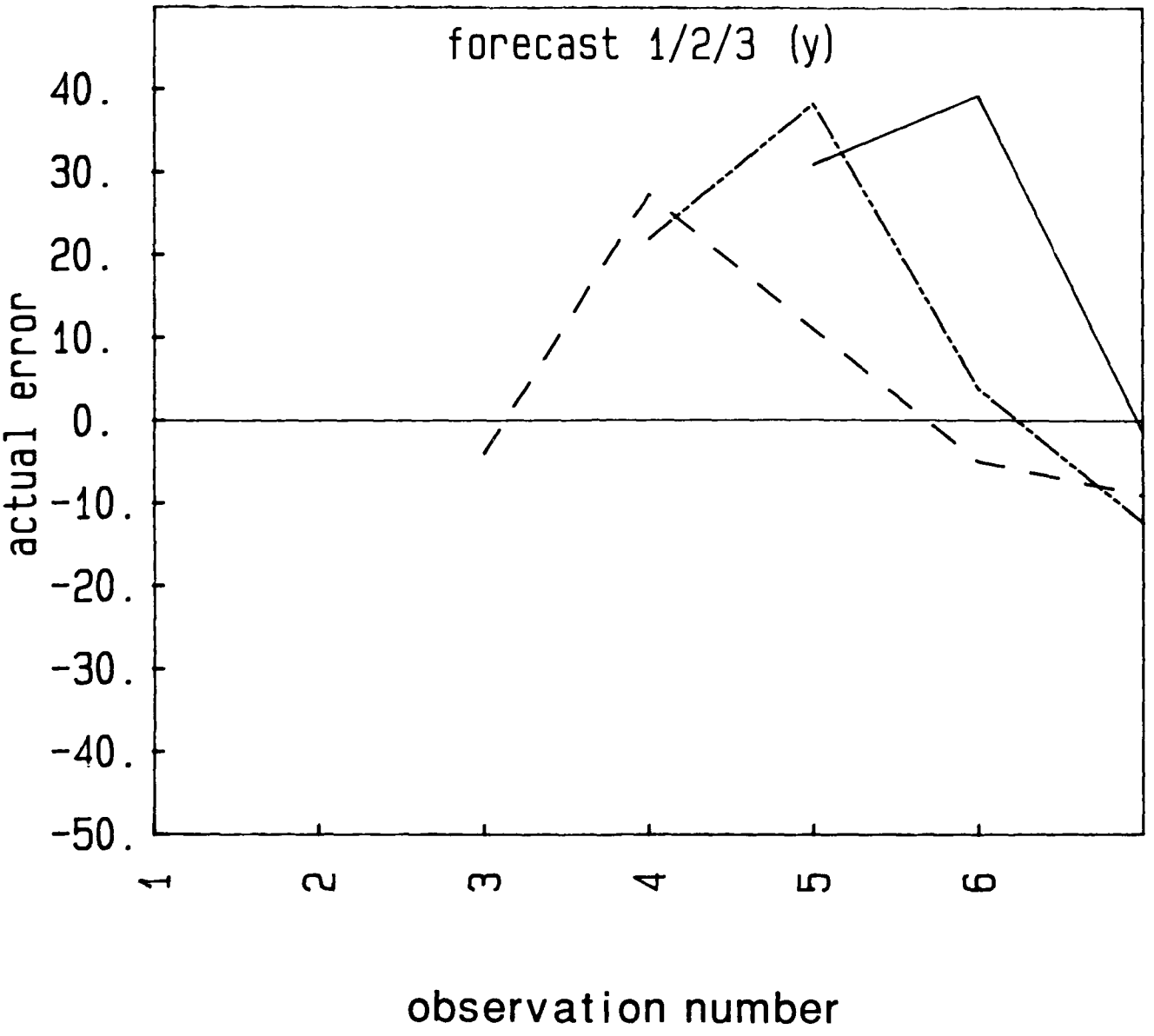

Figure 25b. Plot of forecast error for LINREG method using the position data of Figure 19 for one (dash), two (dash-dot), and three-interval (solid) forecasts: Y position data

behaves poorly with use of the outer contour position data (not shown). On one hand the relatively good performance is somewhat surprising since the LINREG routine weights all data evenly in deriving the best fit trend line. Observation of the outer contour COA data indicated that these data exhibit greater nonlinear behavior than for the inner contour data. Thus, the reasons for success are similar to those previously cited for the KALMAN routine. The LINREG method, by weighting all data equally, tends to preserve the early trends and is less responsive to trend changes than the other nonstationary data routines.

Finally, the actual forecast errors obtained from use of radar precipitation band COA movement shown in Figures 26a and b are presented in Figures 27 - 30 and the accumulated absolute errors are shown in Table 9. These COA data are similar to the IR data, except the data series are significantly shorter and thus emphasize the performance over a short data series. The stationary data routines behave in a manner quite similar to that observed with use of IR data. The ADSIMEXP method is overresponsive and prone to large error, particularly for one-interval forecasts. The KALMAN method is sluggish in response to sudden change. For one-interval forecasts it incurs the greatest error of all nonstationary routines with use of x position data and minimum error with the y position data. The two and three-interval forecast errors are all nearly identical since these forecasts do not incorporate the sudden change at data number 5 and are simply preserving the early trend in the data. The SIMEXP (weight = 0.3) and SIMMOVAV routines respond similarly. The nonstationary data routines providing the best performance are the HOLT2P (weight = 0.7) and BWN1PAD (weight = -0.5) methods. The LINREG method again suffers large error in forecasting one-interval change, again a result of its inability to preferentially weight new observations over the historical trend.

5.4 Computational Complexity

Such tests displayed some relative strengths and weaknesses of the various techniques and narrowed the field of potentially useful routines. Additional factors were then considered. First, ease of implementation is of considerable interest. The computer codes for all the routines have been configured to require a data array transfer between the routine and host program of length no more than ten 32-bit words. The individual routines themselves vary only from about 10-50 lines of code for the simplest to most complex methods. Thus computer code length and data storage considerations are not a useful discriminant.

Another concern is the stability of performance; that is, did routine performance change drastically with introduction of a new data set? The KALMAN and BWNQUAD routines are examples where a complete reversal in relative performance was noted. Because the attributes to be tracked and forecast may behave in unpredictable ways, it is highly doubtful that these two routines would be useful for the broad variation in attribute behavior possible. A related concern is whether the preferred weighting factors, where user-selection was available, varied with introduction of new data. The preferred weight for the HOLT2P method varied between 0.5 and 0.7, whereas the BWN1PAD optimum weight remained at a value of 0.5 throughout all tests. Thus the BWN1PAD method is favored over HOLT2P, since it is not just undesirable, but unacceptable to require determination of an optimum weight while simultaneously deriving forecasts. The ADSIMEXP method attempts to make such an automatic adjustment, however, it continually overreacted to data trend variations. The SIMMOVAV and LINMOVAV routines appear to offer middle of the road performance, while not

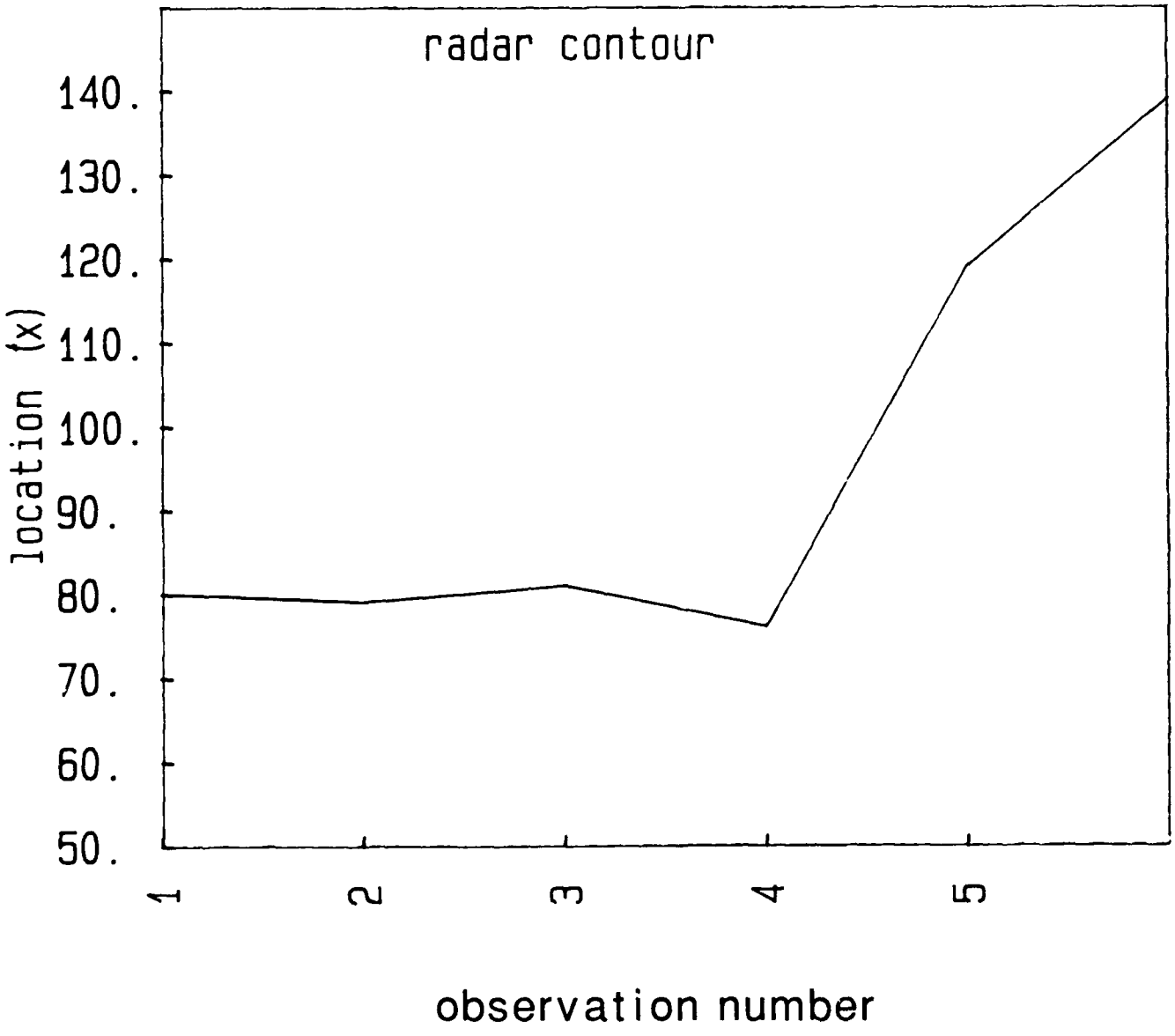

Figure 26a. Plots of center of area (COA) image pixel locations for X (east) direction for 32 dBZ contour area from radar observation of Hurricane Gloria

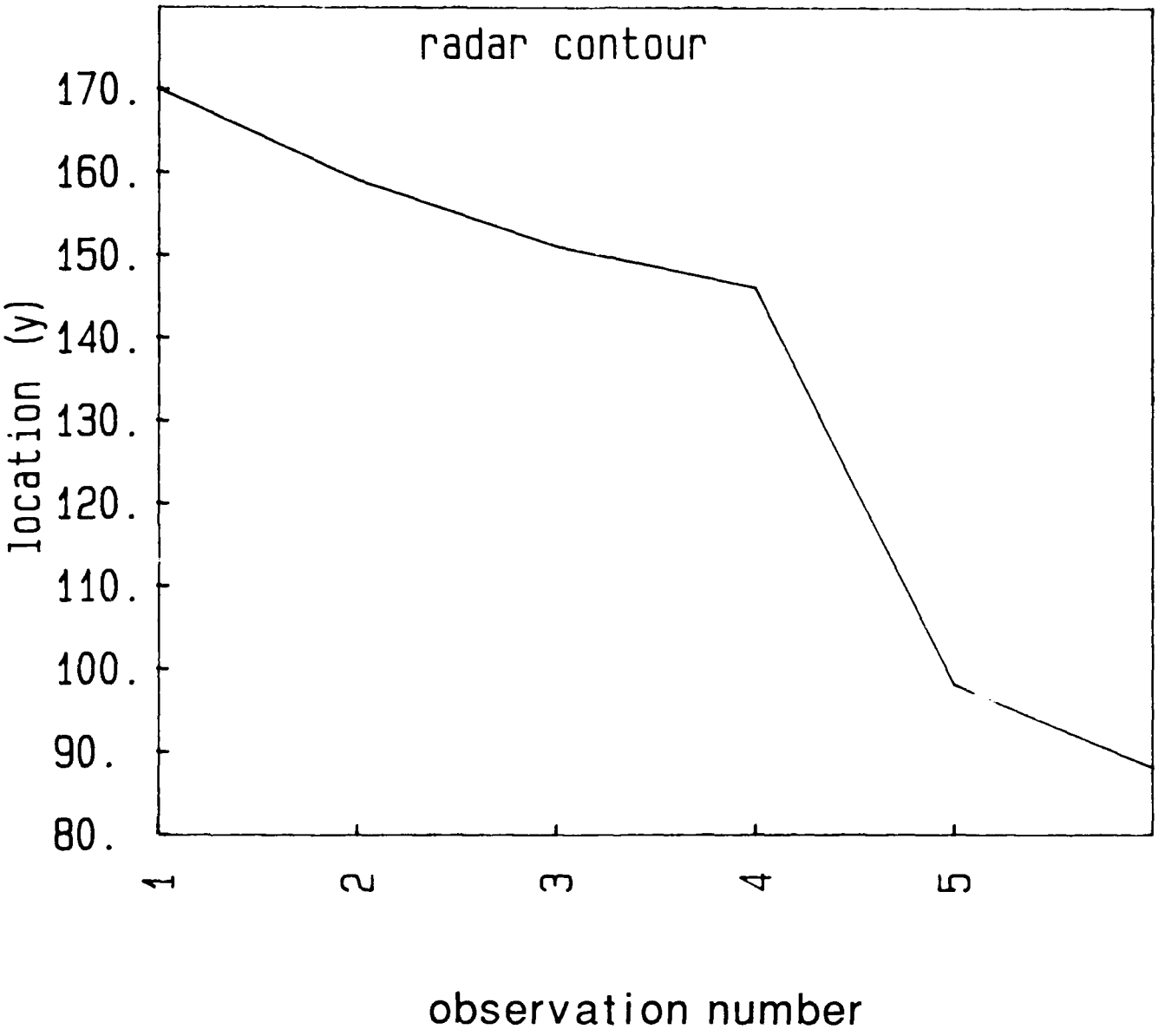

Figure 26b. Plots of center of area (COA) image pixel locations for Y (north) direction for 32 dBZ contour area from radar observation of Hurricane Gloria

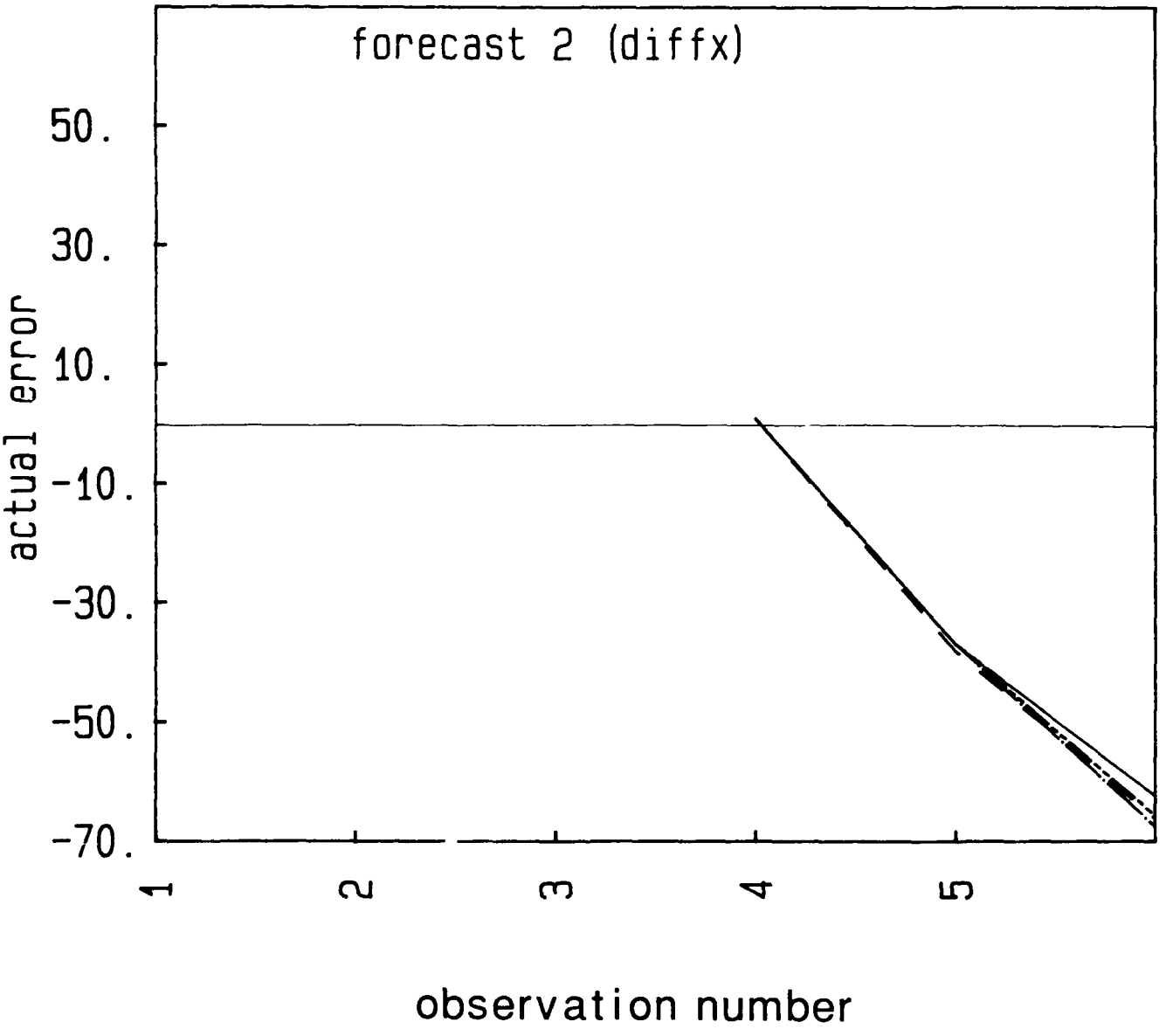

Figure 27a. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 26 for one-interval forecasts

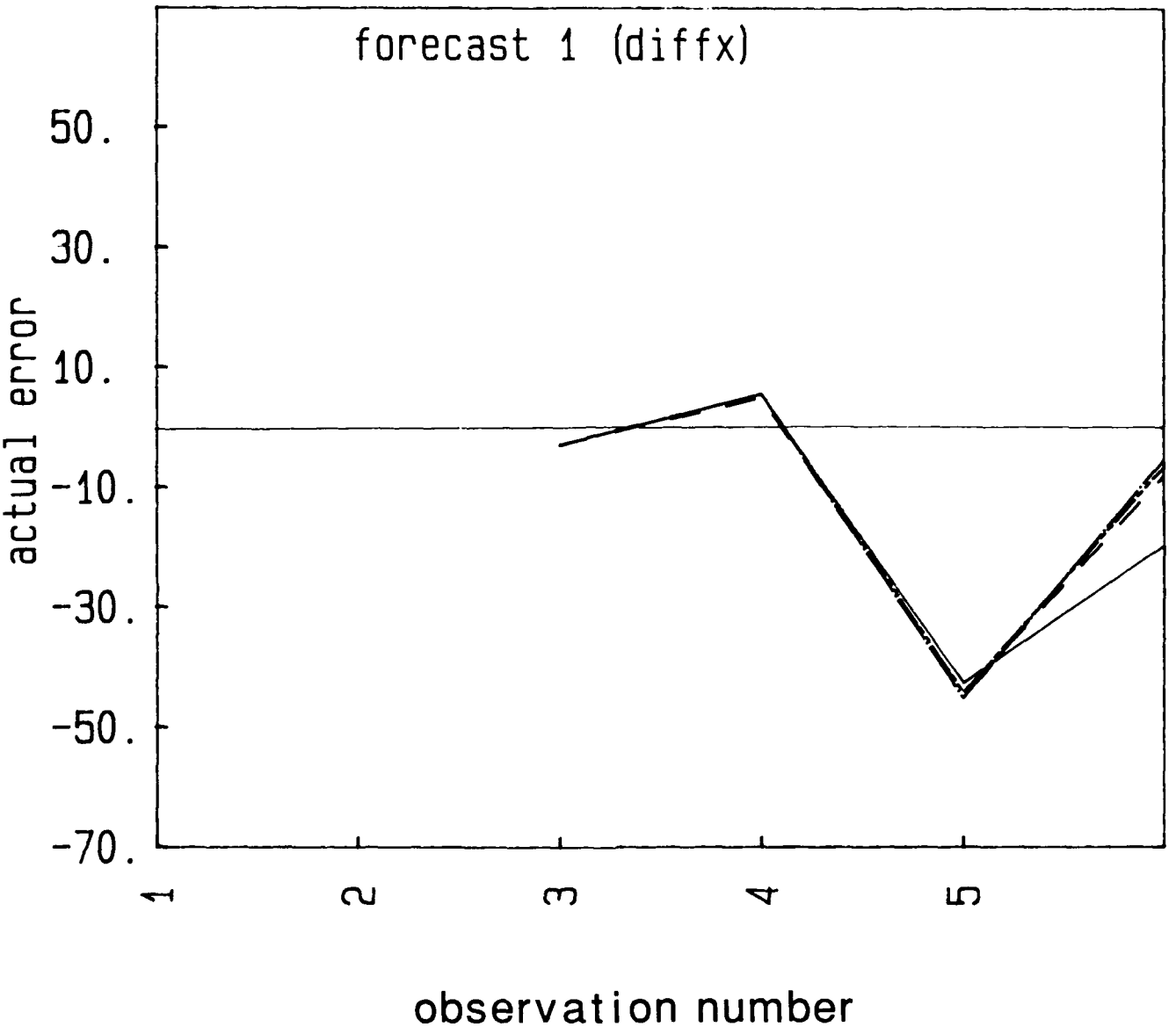

Figure 27b. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 26 for two-interval forecasts

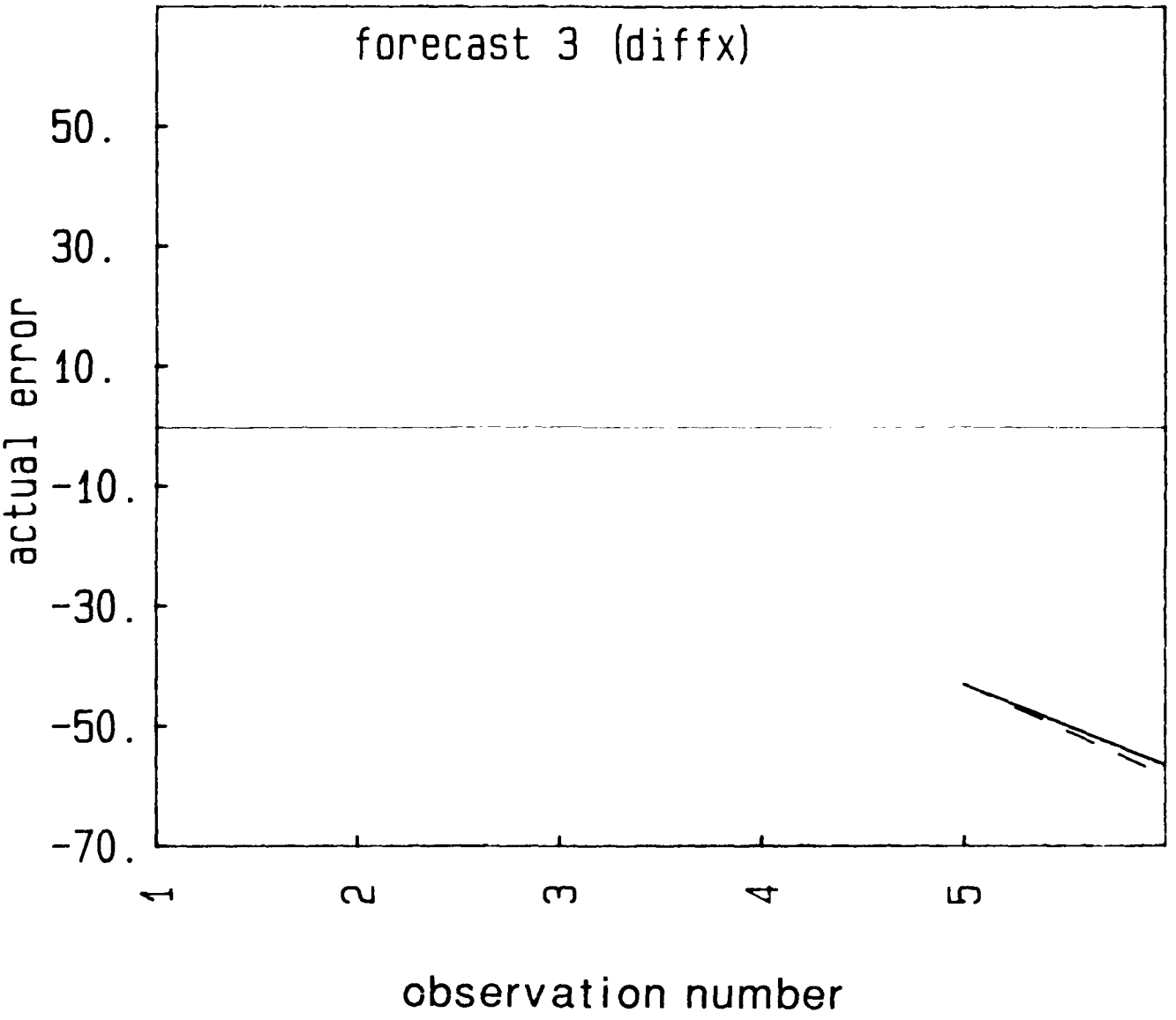

Figure 27c. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of X location data of Figure 26 for three-interval forecasts

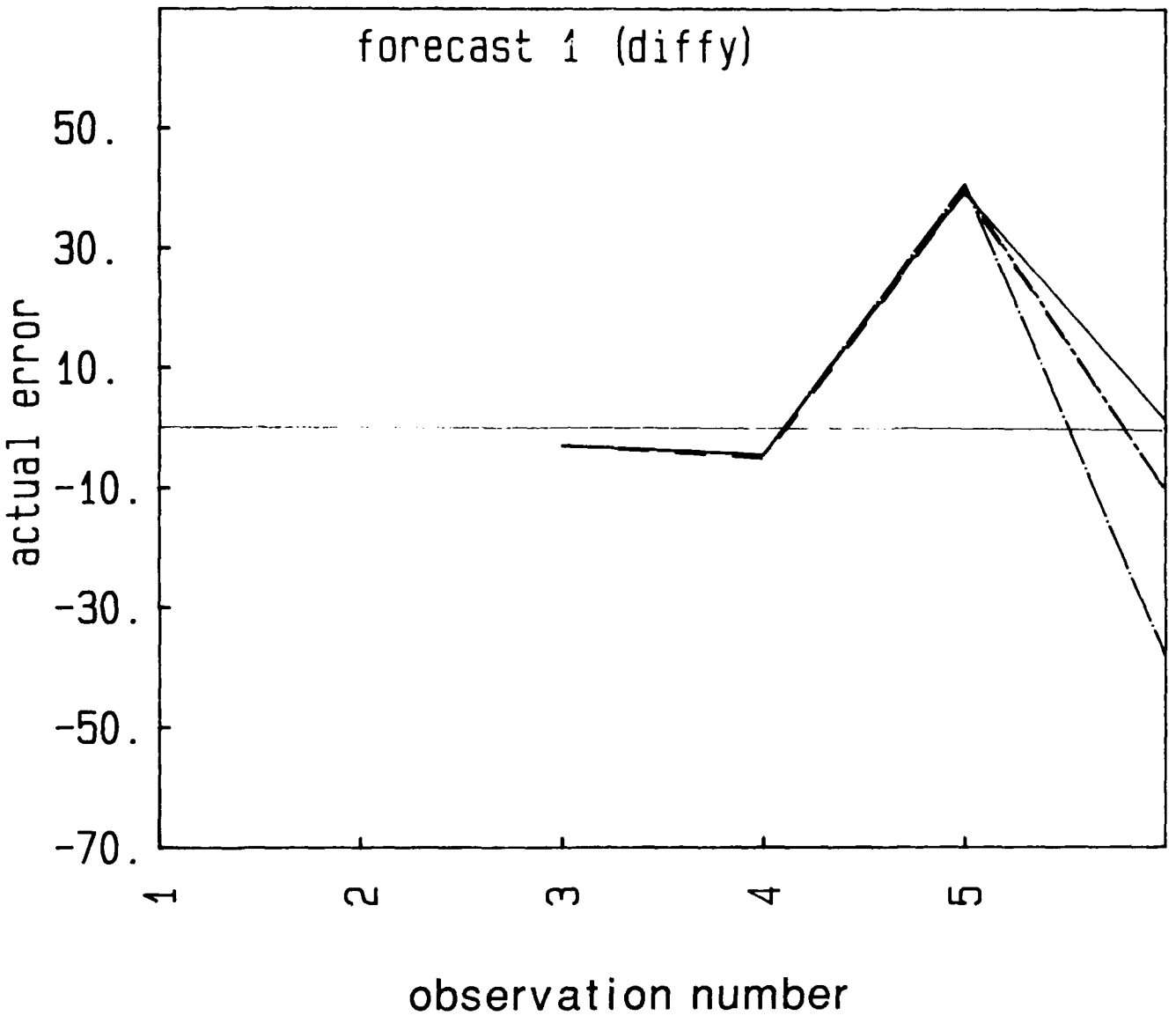

Figure 28a. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 26 for one-interval forecasts

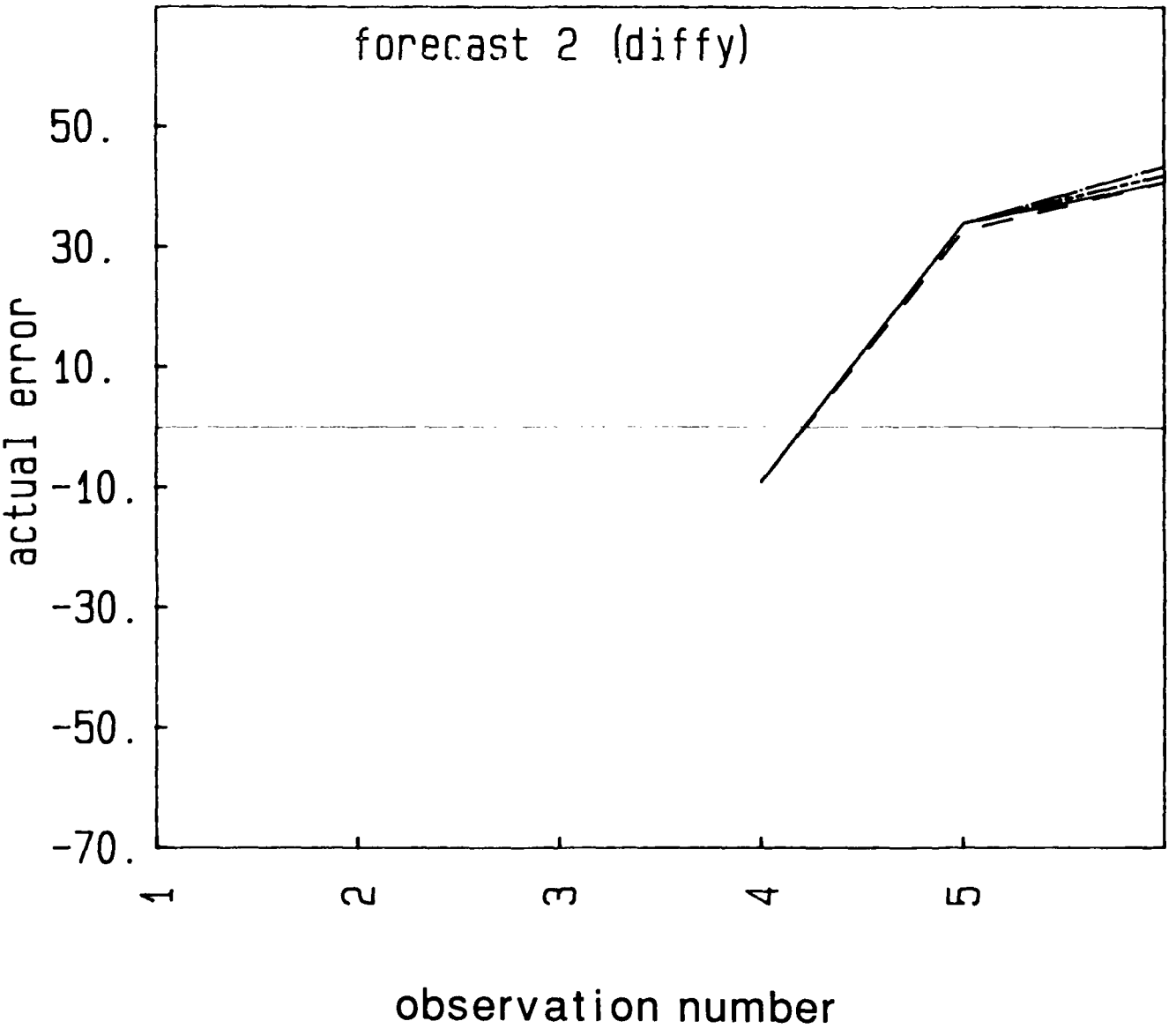

Figure 28b. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 26 for two-interval forecasts

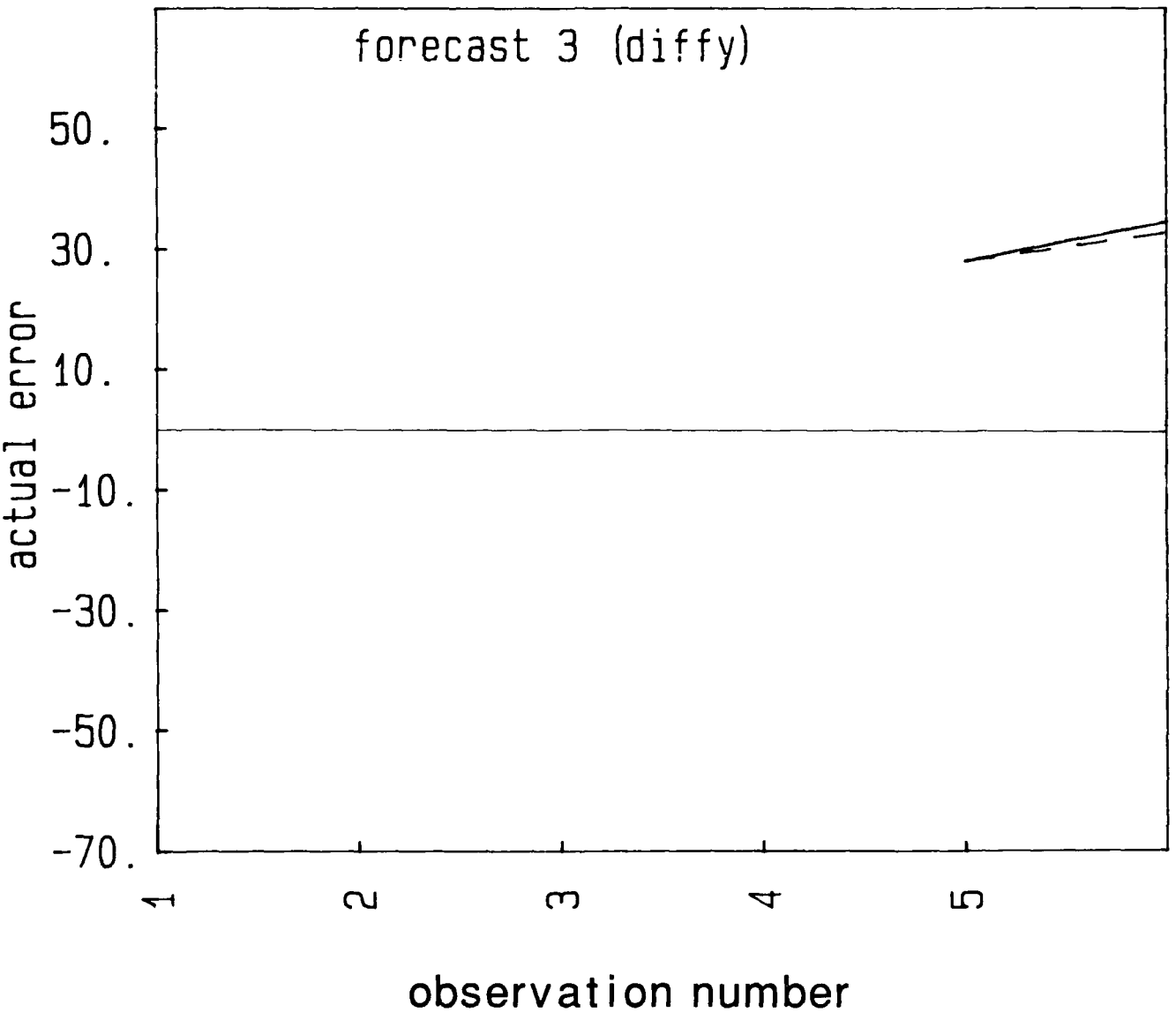

Figure 28c. Plots of actual forecast error for SIMEXP (dash), ADSIMEXP (dash-dot), SIMMOVAV (long/short dash), KALMAN (solid) routines using differences of Y location data of Figure 26 for three-interval forecasts

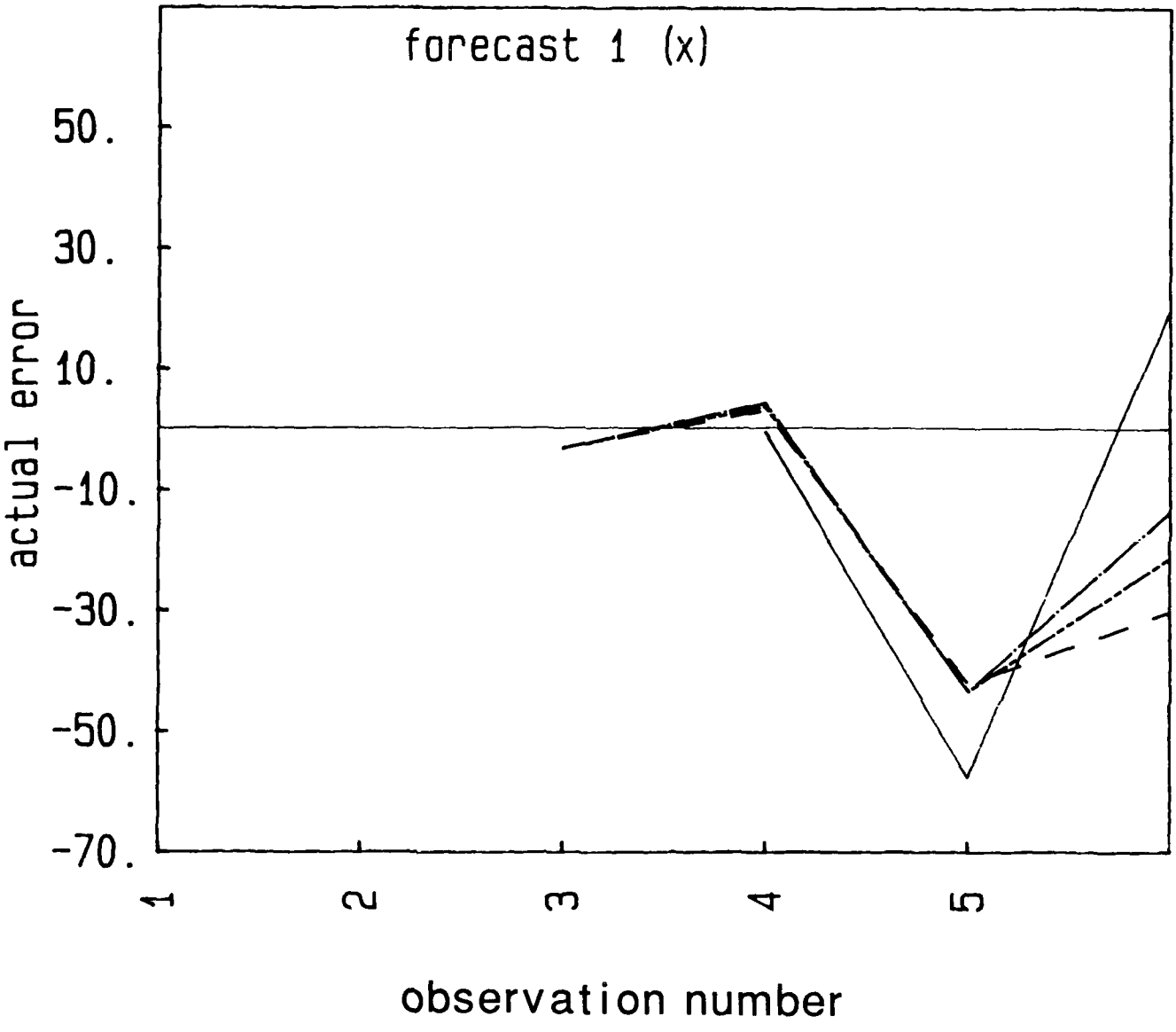

Figure 29a. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 26 for one-interval forecasts

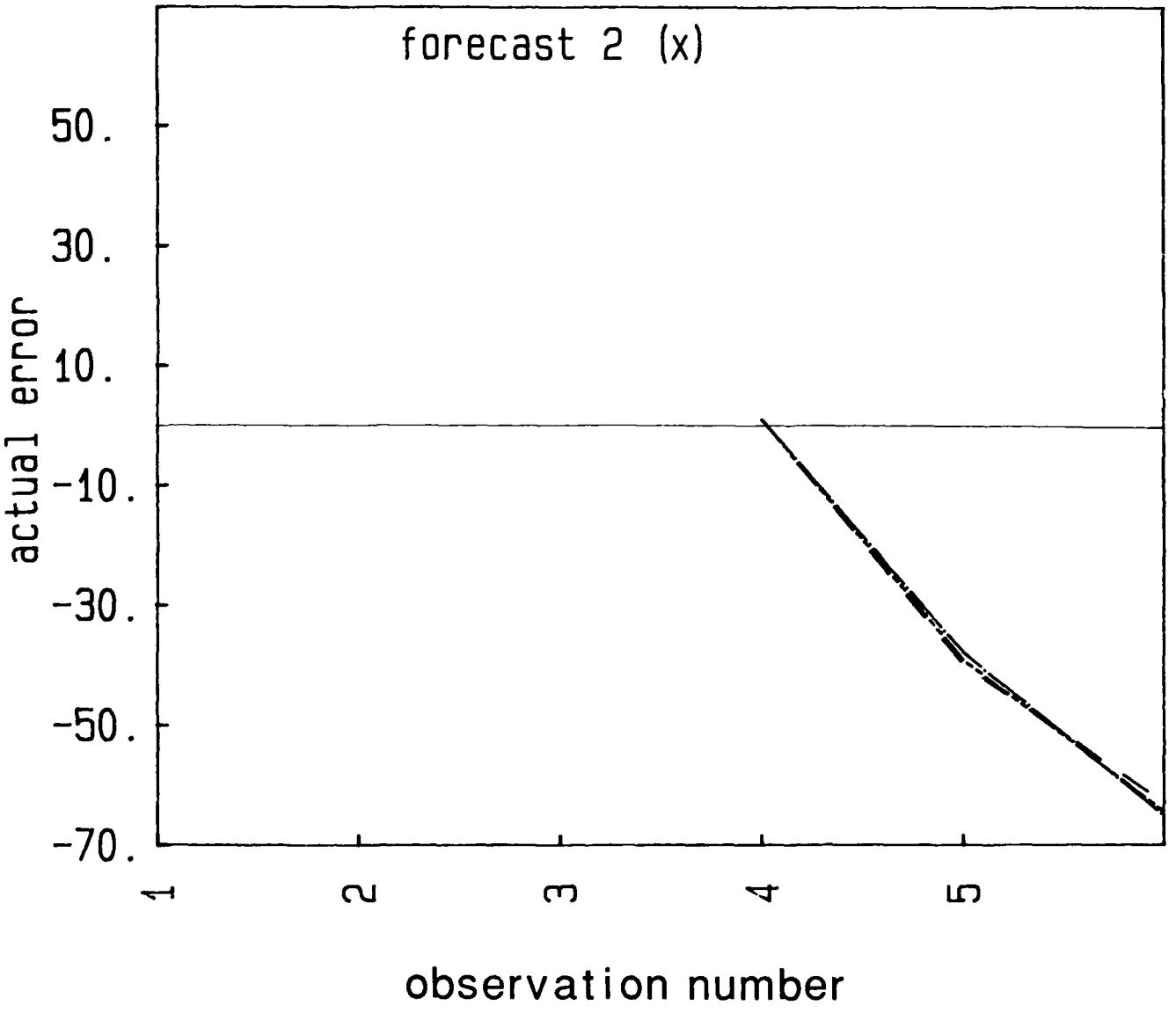

Figure 29b. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 26 for two-interval forecasts

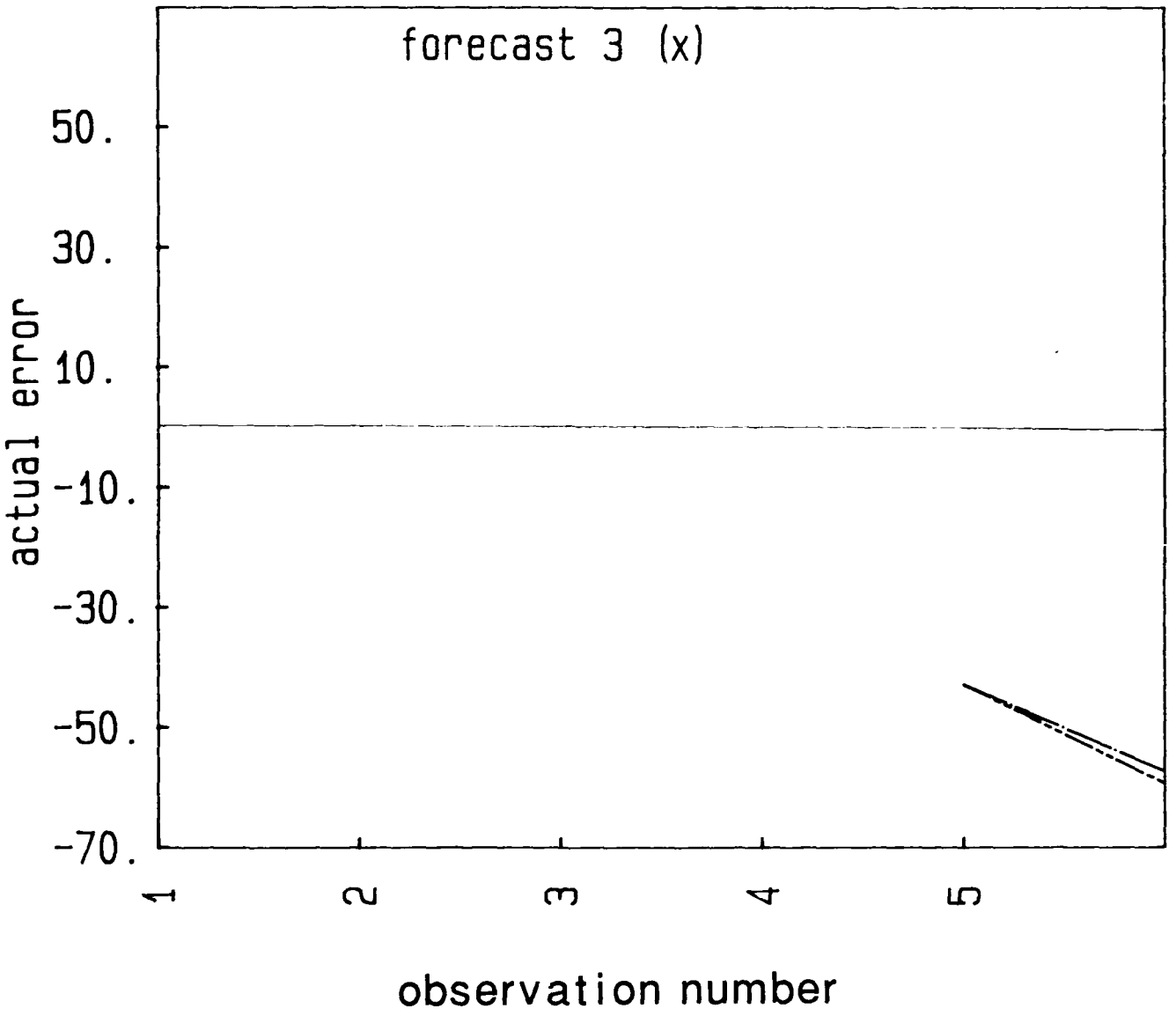

Figure 29c. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using X location data of Figure 26 for three-interval forecasts

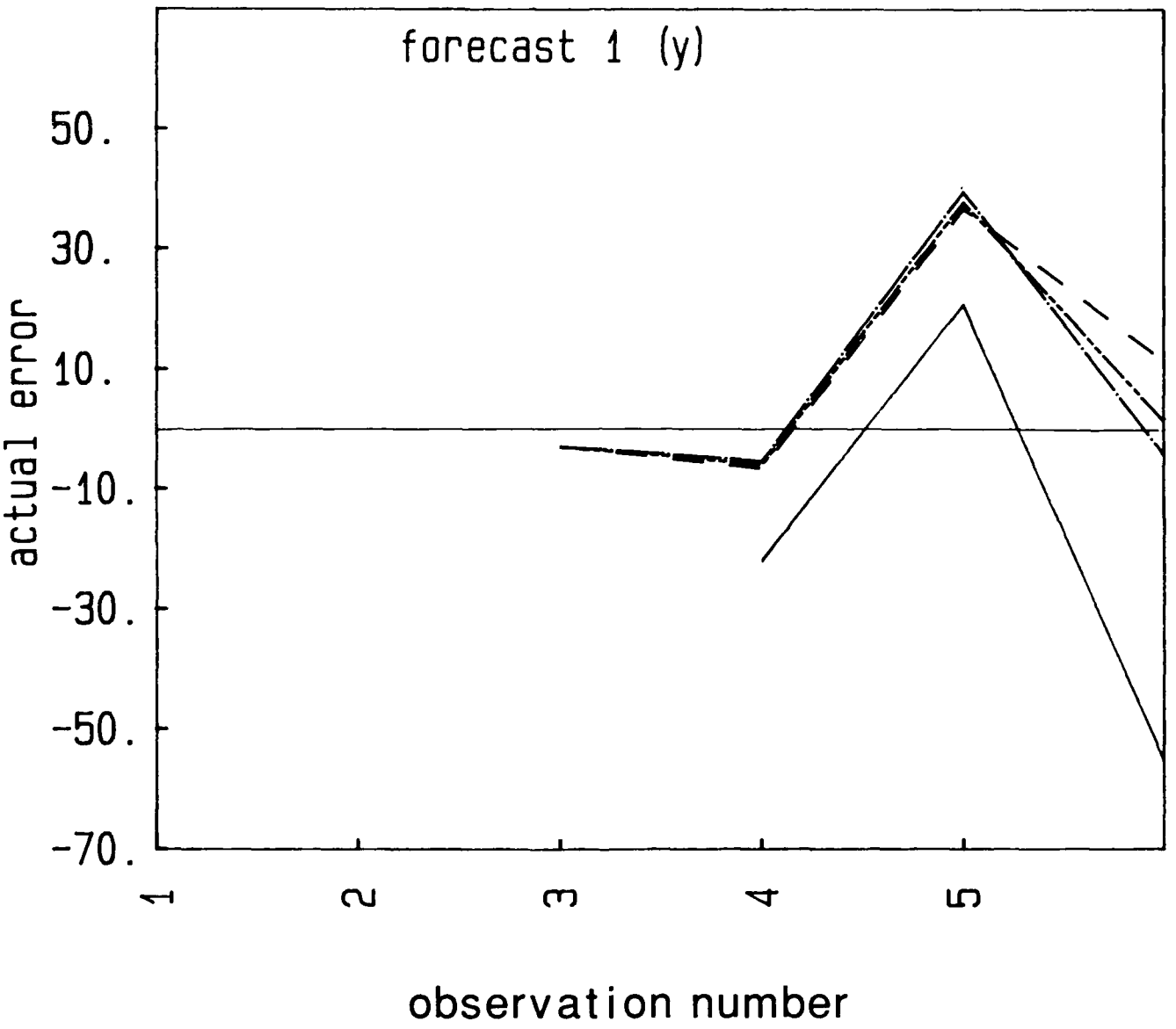

Figure 30a. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 26 for one-interval forecasts

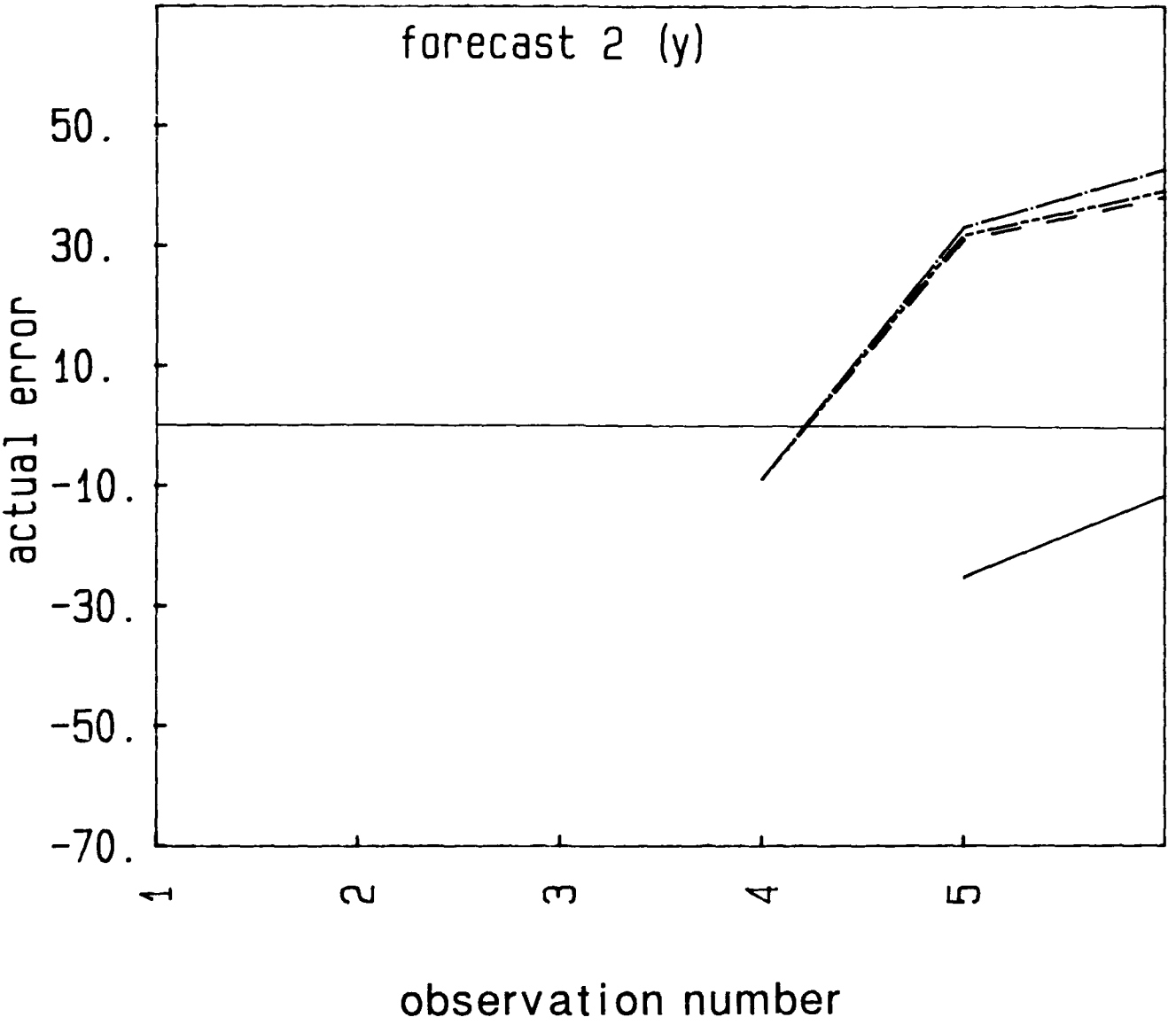

Figure 30b. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 26 for two-interval forecasts

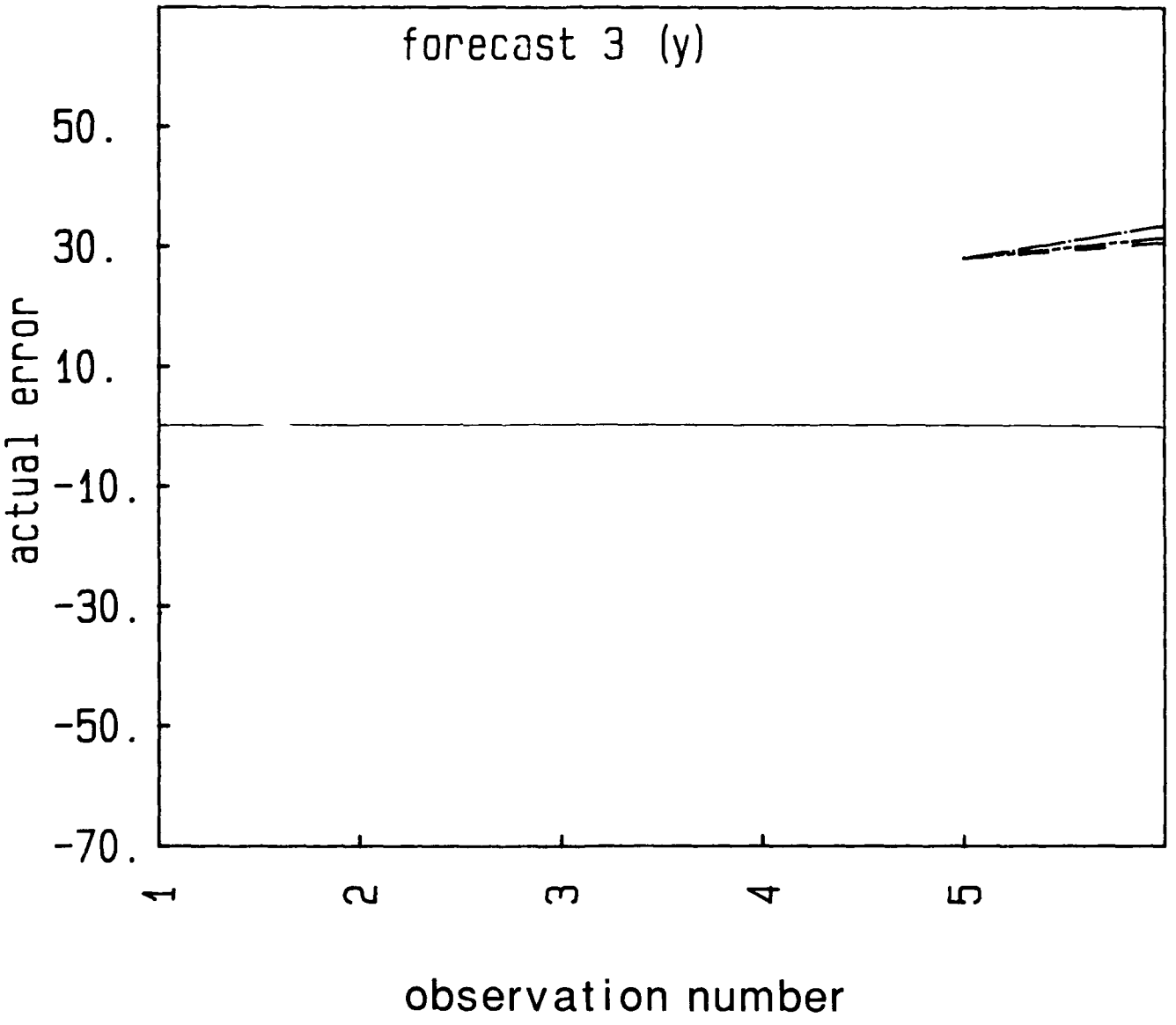

Figure 30c. Plots of actual forecast error for LINMOVAV (dash), HOLT2P (dash-dot), BWN1PAD (long/short dash), BWNQUAD (solid) routines using Y location data of Figure 26 for three-interval forecasts

Table 9. Accumulated absolute forecast error obtained from application of Center of Area data from radar rainband observations from Hurricane Gloria with the forecasting techniques. Stationary (nonstationary) techniques employ difference (original) data. Results for techniques allowing user-selection of weights are shown for the three weight values W = 0.3, 0.5, and 0.7.

DATA	SIMEXP			ADS IMEXP	SIMM OVAV	KAL MAN	LINM OVAV	LIN REGR	HOLT2P			BWNIPAD			BWNQUAD		
TYPE	WT .3	.5	.7						.3	.5	.7	.3	.5	.7	.3	.5	.7
	← POSITION DIFFERENCE DATA →						← ORIGINAL INPUT POSITION DATA →										
	FIRST FORECAST INTERVAL																
X	60.7	54.1	64.4	59.2	59.5	70.9	78.6	80.0	94.4	90.6	64.5	57.0	71.5	86.4	55.1	68.3	77.6
Y	57.9	65.9	74.2	86.3	57.8	48.2	57.6	56.2	63.1	59.4	52.3	61.5	48.0	58.3	207.8	111.9	98.0
	SECOND FORECAST INTERVAL																
X	105.3	105.5	106.1	105.5	103.7	100.5	104.0	102.0	105.9	106.3	104.1	105.0	104.8	105.5	33.3	64.2	159.4
Y	82.7	86.5	89.9	86.5	85.0	83.8	78.0	83.0	70.9	70.8	84.7	85.8	79.8	73.4	288.9	155.9	37.1
	THIRD FORECAST INTERVAL																
X	101.3	99.5	97.7	99.5	99.5	99.5	103.3	100.0	105.3	105.5	100.5	99.9	102.5	104.7	17.2	5.3	102.2
Y	60.7	62.5	64.3	62.5	62.5	62.5	58.7	62.0	56.7	56.5	61.5	62.1	59.5	57.3	208.7	135.1	90.1

providing for the flexibility offered in user-selected weighting schemes. The LINREG performed better than expected, rivaling the BWN1PAD method on numerous occasions. Although it responds slowly to new data trends it does easily accept non-uniformly spaced data. This last point should not be taken lightly, for the other routines currently assume uniformly spaced data. Although an aperiodic data series can be modified to develop a uniform time base, this would require some additional computer code and calculations.

The efficiency, or speed, of the various routines is also of vital concern. If a large number or parameters are to be tracked and forecast, then the fastest routine, assuming comparable accuracy, will best support the real-time operational mode. With the routines under consideration, speed effectively translates into the number of mathematical operations required to make a simple forecast. The numbers of operations, utilizing the exact algorithm after reception of an updated data value are shown in Table 10 for the more reasonable techniques (number of computations to perform data differencing for stationary routines are included in cited numbers).

The SIMMOVAV and SIMEXP techniques, even employing difference data, are at least twice as fast as the HOLT2P and BWN1PAD techniques on original data, and at least five times as fast as the LINREG and KALMAN methods. After taking into consideration all factors the SIMEXP and BWN1PAD methods were chosen as the primary forecast routines. It is quite likely that a two-tiered approach, where both routines would be available for use could be employed. This would require monitoring either the variation (variance, trend, etc.) of the data being received, or the forecasts being developed, in order for real-time selection to be made. Finally, the LINREG method was also retained in the event it became necessary to employ aperiodic time series data.

5.5 Reconstruction

The contour segments derived from the forecast attributes must be joined together to form a continuous and meaningful physical contour. The methods under consideration, as demanded by the desire to obtain real-time operation, are simple and quick. Only two cases need be considered. namely, when successive segments do not touch and when the segments overlap. In the overlap case, the simple rule is simply to discard those portions of segments beyond the overlap point which would not be part of a continuous boundary. For the non-intersecting case, the object is to fit a line segment between the end points. This can be accomplished by: (1) extending the segments on both sides until an intersection is obtained, (2) fitting a polynomial across the gap, or (3) fitting a straight line segment across the gap if the segment would be short and the included angles between the new segment and the original segments are near 180 degrees.

6. CONCLUSIONS

A methodology has been developed for the short-term forecasting of cloud and precipitation fields. It employs pattern recognition techniques to extract the useful features from the data field and extrapolative techniques to project these features into the future. The methodology also relies heavily on the use of data reduction techniques that significantly reduce the amount of information that must be tracked and forecast. The methodology is generic in the sense that it may be applied to any data

Table 10. The number of computations to generate a new forecast given a new observation. Stationary techniques employ difference data and all routines utilize exact technique formulations.

OPERATIONAL COMPLEXITY

TECHNIQUE NAME	NUMBER OF OPERATIONS	
	*/-	+/-
SIMMOVAV	1	4
SIMEXP	2	4
BWNIPAD	5	7
HOLT2P	5	7
LINREG	10	8
KALMAN	11	24

which can be represented as a two-dimensional distribution. For the 0 - 0.5 hr forecast problem addressed here, the primary data source is radar reflectivity factor. Use of GOES satellite data, having a temporal scale comparable to the maximum forecast interval, could be employed as an indicator of the overall large scale environment, but is not incorporated in the quantitative forecasting methods.

The methodology is broken into the four component steps of (1) data preprocessing, (2) feature definition and extraction, (3) attribute definition and change determination, and (4) attribute forecasting and feature reconstruction. The techniques employed within each step were strongly influenced by the desire to develop a real-time capability.

The features tracked are contours of radar reflectivity factor, or equivalently, precipitation intensity. The contour data, on horizontal grids at selected heights, are composited fields obtained either by averaging vertical segments or by taking the maximum value above each grid point. The contour data are smoothed by median and lowpass filters to remove data gaps, errors, and noise.

Initial methods considered for describing whole contours were shown to be inadequate for accurate measurement of preferential evolution and displacement along the contours. The processes of feature definition and extraction focuses on two primary algorithms for segmentation of contours. One method involved objectively segmenting the contour into straight line segments of varying length. A second method employed fixed length segments, typically 10 - 25 percent of the entire contour length. The individual segments are described in terms of attributes, a highly limited set of quantities that accurately describe each segment. Examples of attribute quantities include segment length, starting location of the segment, orientation, and distance from some global reference location.

The evolution of features was determined by monitoring the changes in the segment attributes between successive observations and developing a time history of these values. These data form the data base to be used by the forecasting process.

A number of forecasting techniques were investigated by observing their quantitative and qualitative performance with limited sets of generated and real test data. The generated data included simple time series representing stationary data with added noise, and noise-free linear and quadratic trends. The real test data were derived from radar and satellite observations of features associated with Hurricane Gloria. Tested techniques ranged from simple moving filters to triple smoothing quadratic filters. It was determined that estimating quadratic trends from data where erratic and significant changes could occur could easily defeat the quadratic method. Techniques that automatically determine new weights for current and past observations are found to either overreact (Adaptive Exponential Smoothing filter) or underreact (Kalman filter) to these sudden changes. Ultimately two techniques were selected: (1) the Simple Exponential Smoothing filter, and (2) the Brown One-Parameter Adaptive Smoothing filter. Whereas the exponential filter would operate on a time series obtained from the differences of the attributes between observations, the Brown filter operates on the original attribute observations. These routines worked well with stationary and linear time series data. More complicated trends are handled fairly well by these routines due to their ability to preferentially weight the most recent observations. From the time series of attribute data, the forecasting process develops future expectations of attribute values. These values completely determine future contour segments, and ultimately the future contours.

The software developed represents an extensive library of routines. Currently, a major portion of these routines are accessed from an executive program through interactive menu-driven operations.

Further testing of the segment (line and curve fitting) routines is required to fine tune these techniques. The process of reconstruction of the forecast segments to produce accurate and pleasing future contours has not been fully completed. Also, this software package does not currently facilitate real-time operation, a definite initial objective of the effort. These efforts of fine tuning existing routines and development of an executive program for real-time operation will be accomplished in a new more comprehensive short-term forecasting program to begin shortly.

References

1. Sadoski, P.A., Egerton, D., Harris, F.I, and Bohne, A.R. (1988) *The Remote Atmospheric Probing Information Display (RAPID)*, AFGL-TR-88-0036, AD A196314.
2. Bohne, A.R. and Harris, F. Ian (1985) *Short Term Forecasting of Cloud and Precipitation*, AFGL-TR-85-0343, AD A169744.
3. Mohr, C.G. and Vaughan, R. (1979) An economical procedure for Cartesian interpolation and display of reflectivity factor data in three-dimensional space, *J. Appl. Meteorol.* **18**:661-670.
4. Foote, G.B. and Mohr, C.G. (1979) Results of a randomized hail suppression experiment in northeast Colorado. Part VI: Post hoc stratification by storm intensity and type, *J. Appl. Meteorol.* **18**:1589-1600.
5. Harris, F.I. and Petrocchi, P.J. (1984) *Automated Cell Detection as a Mesocyclone Precursor Tool*, AFGL-TR-84-0266, AD A154952, 32 pp.
6. Pratt, William K. (1978) *Digital Image Processing*, Wiley-Interscience, New York.
7. Rinehart, R.E. (1979) *Internal Storm Motions from a Single Non-Doppler Weather Radar*, NCAR/TR-146+STR, National Center for Atmospheric Research, Boulder, CO, 262 pp.
8. Smythe, G.R. and Zrnic', D.S. (1983) Correlation analysis of Doppler radar data and retrieval of the horizontal wind, *J. Clim. Appl. Meteor.* **22**:297-311.
9. Smythe, G.R. and Harris, F.I. (1984) *Sub-Cloud Layer Motions from Radar Data Using Correlation Techniques*, AFGL-TR-84-0272, AD A156477.
10. Dudani, S.A. (1976) Region extraction using boundary following, *Pattern Recognition and Artificial Intelligence* (C.H. Chen, editor), Academic Press, Inc., New York, NY, pp. 216-232.
11. Clodman, S. (1984) Application of automatic pattern methods in very-short-range forecasting, *Proc. Nowcasting II Symposium*, Norrkoping, Sweden, ESA SP-208.
12. Williamson, D.L. and Temperton, C. (1981) Normal mode initialization for a multilevel grid-point model, Part II: Nonlinear Aspects, *Mon. Wea. Rev.* **109**:744.
13. Dennis, J.E., Gay, D.M., and Welsch, R.E. (1977) *An Adaptive Nonlinear Least-Squares Algorithm*, Cornell Computer Science TR77-321.
14. Freeman, H. (1961) On the encoding of arbitrary geometric configurations, *IRE Trans. Electron. Comput.*, **EC-10**:260-269.

15. Wu, Li-De (1982) On the chain code of a line, *IEEE Trans. Pat. Anal. and Mach. Intel.*, **PAMI-4** (#3):347-353.

Appendix A

Forecasting Algorithms

Two specific algorithms were selected for use in the RAPID methodology for deriving forecasts of attribute variables. These were the Simple Exponential Filter and the Brown's One-Parameter Linear Exponential Filter. The particular formulations are presented in the following sections.

A1. SIMPLE EXPONENTIAL FILTER

This filter is designed to operate on stationary data. The data is assumed composed of two components, a population mean and added noise. The form of the filter is

$$F(t+1) = W\,X(t) + (1-W)\,F(t) \qquad (A1)$$

$$F(t+1) = W\,X(t) + W\,(1-W)\,X(t-1) \ldots W\,(1-W)^{**}N-1\,X(t-(N-1)) \qquad (A2)$$

$$F(t+1) = F(t) + W\,(x\,(T) - F(t)) \qquad (A3)$$

in its various forms. Here F(t) is the previous forecast valid for the current time, F(t + 1) is the current forecast for the next observation period, X(t) is the current observation, and W is a user-selectable weighting factor. The formulation (A1) requires only storage of the current forecast and the weighting factor while still empoying all data ever input into the filter (A2). As shown in (A2) a small value of W effectively results in near uniform weighting of all data ever input into the filter, whereas a large weight value biases the filter result towards current observations. In this manner the filter can be biased to the historical mean (low signal to noise ratio) or can adjust to new trends (high signal to noise ratio). This adjustment feature is shown more clearly in (A3) which may be interpreted as a new

forecast being derived from the previous forecast and an adjustment proportional to the error incurred from the previous forecast.

Because of its simple structure and ease of implementation, this filter may efficiently be used with difference data as input, where the input series is now the differences between successive observations. This allows application where the forecast variable may contain a linear trend. The results with test data indicated that preferential weighting should be given to the gistorical triend to diminish noise effects and overestimation of the rate of change. A value of W = 0.3 was found most suitable.

A2. BROWN ONE-PARAMETER LINEAR EXPONENTIAL FILTER

This filter is designed for use with data exhibiting an underlying linear trend. It performs a number of smoothing operations to obtain estimates of the linear equation parameters and filter out the random noise component. This technique may be written as

$$F(t\ m) = A(t) + B(t)\ m \tag{A4}$$

where

$$A(t) = S1(t) + (S1(t) - S2(t)) \tag{A5}$$

$$B(t) = (S1(t) - S2(t))\ W/(1-W) \tag{A6}$$

$$S1(t) = W\,X(t) + (1-W)\,S1(t-1) \tag{A7}$$

$$S2(t) = W\,S1(t) + (1-W)\,S2(t-1) \tag{A8}$$

where F(t + m) is the forecast for m intervals from the current (t) time, X(t) is the current observation, W is a user-selected weight, and A(t), B(t) are the automatically adjusting parameters of the linear trend equation. Setting W large encourages the filter to preferentially adjust the coefficients of the linear equation to the new observations rather than the historical data. This effectively makes the assumption that noise is a minor contribution and that trend changes suggested by new observations should be heavily weighted. If the signal to noise ratio is expected to be large and heavy smoothing is required to bring out the trend then a smaller weight must be applied. The test data suggested that a value of W = 0.5 would accommodate a wide range of data types and is the value employed in RAPID.

•U.S. GOVERNMENT PRINTING OFFICE:1989-600-000/00018