

PL-TR-91-2008

AD-A240 510



Mesoscale Prediction and Satellite Cloud Analysis
for Advanced Meteorological Processing Systems

Gary B. Gustafson
Jean-Luc Moncet
Charles F. Ivaldi
Ho-Chun Huang
Jeanne M. Sparrow



Atmospheric and Environmental Research, Inc.
840 Memorial Drive
Cambridge, Massachusetts 02139

8 January 1991

"Original contains color
plates: All DTIC reproductions
will be in black and
white"

Final Report
Period Covered April 1988 - December 1990

— Approved for public release;
Distribution unlimited —



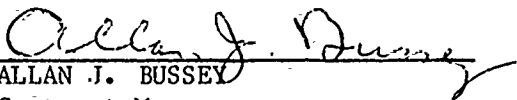
PHILLIPS LABORATORY
AIR FORCE SYSTEMS COMMAND
HANSCOM AIR FORCE BASE, MASSACHUSETTS 01731-5000

91 9 18 023

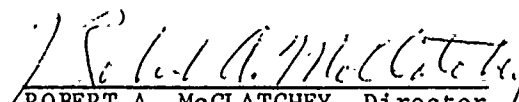
91-10971



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


ALLAN J. BUSSEY
Contract Manager

FOR THE COMMANDER


ROBERT A. McCLATCHEY, Director
Atmospheric Sciences Division

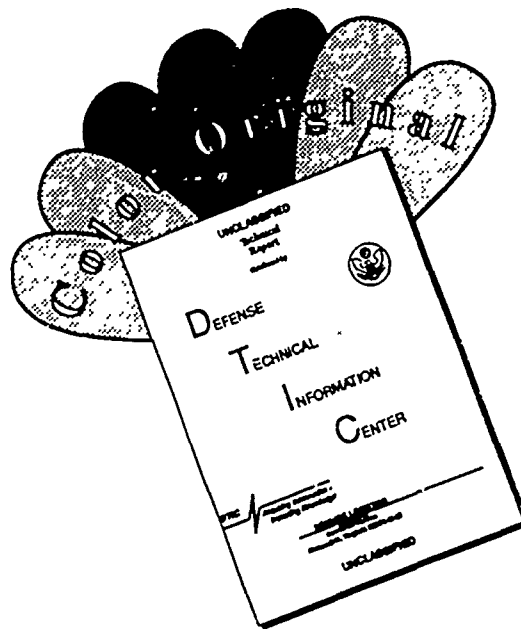
This report has been reviewed by the ESD Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

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.

Do not return copies of this report unless contractual obligations or notices on a specific document requires that it be returned.

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF COLOR PAGES WHICH DO NOT REPRODUCE LEGIBLY ON BLACK AND WHITE MICROFICHE.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 8 January 1991	3. REPORT TYPE AND DATES COVERED Final Report April 1988-December 1990	
4. TITLE AND SUBTITLE Mesoscale Prediction and Satellite Cloud Analysis for Advanced Meteorological Processing Systems			5. FUNDING NUMBERS FE: 62101F PR 6670 TA 17 WU BA Contract F19628-88-C-0086	
6. AUTHOR(S) Gary B. Gustafson Charles F. Ivaldi Jeanne M. Sparrow Jean-Luc Moncet Ho-Chun Huang				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Atmospheric and Environmental Research, Inc. 840 Memorial Drive Cambridge, MA 02139			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Phillips Laboratory (LY) Hanscom AFB, MA 01731-5000 Contract Manager: A. Bussey/LY			10. SPONSORING/MONITORING AGENCY REPORT NUMBER PL-TR-91-2008	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report summarizes work that was performed under a research support contract with the Atmospheric Prediction and Satellite Meteorology branches of the Geophysics Laboratory Atmospheric Sciences Division. Three topics are addressed: mesoscale prediction modeling, improvements and modifications to the Air Force real-time nephanalysis program, and on-site support for maintenance and enhancements to the Air Force Interactive Meteorological Processing System (AIMS). Mesoscale model work centered on the GL/LYP prediction model developed by the Aster Corporation. An investigation of the model revealed several problems and inconsistencies when attempting to reproduce the Aster results. Numerous procedural changes were made to the model program to facilitate operation and interpretation of model results. Nephanalysis work was in a number of areas including: satellite data ingest and data base management, remapping of satellite data, and development of an atmospheric transmission band model to characterize infrared water vapor attenuation. Band model results were inconclusive due to model sensitivity to input data field quality, particularly surface temperature and vertical moisture profiles. AIMS support consisted of system management, technical support, and software development over a time of explosive system growth. During the period of the contract, 11 processing nodes were added to the cluster in addition to numerous peripheral devices and software upgrades.				
14. SUBJECT TERMS Nephanalysis Satellite meteorology Mesoscale modelling Data base management			15. NUMBER OF PAGES 146	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT SAR	

TABLE OF CONTENTS

	Page
1. Introduction	1
2. Investigation of the GL Mesoscale Forecast Model	3
2.1 Model Characteristics	3
2.2 Model Run-Time Environment	8
2.2.1 Preprocessor	9
2.2.2 Job Control File	11
2.3 Display of Model Results	13
2.3.1 Model Output	13
2.3.2 Reading Stored Data	14
2.3.3 Graphical Display	15
2.4 Comparison Results	23
2.5 Summary	24
3. Nephanalysis Data Base	30
3.1 NDB Data Base Structure	31
3.1.1 AFGWC Polar Stereographic Grid System	33
3.2 NDB Run-Time Library	34
3.3 NDB Utility Program	34
4. RDNEPH Data Ingest Utilities	36
4.1 Satellite Global Data Base	36
4.2 Upper Air Temperature and Humidity	38
4.3 Surface Temperature	39
4.4 Background Brightness	40
4.5 Terrain Heights and Geography Type	41
5. AVHRR Mapping Utility	43
5.1 AVHRR Data	46
5.2 RTNEPH Grid	48
5.2.1 Coordinate Transformation Equations	50
5.3 Calculation of Ephemeris Data	51
5.4 Search Methods, Nearest Neighbor Criterion	52
5.4.1 The Binary Search Methods	55
5.4.2 The Random Search Method	56
5.5 Mapping Considerations	57
5.6 Data Processing Considerations	58
5.7 Unsolved Problems and Possible Solutions	66
5.8 Summary	67



Availability Codes	
Dist	Avail and/or Special
A-1	

TABLE OF CONTENTS

(continued)

	Page
6. A New Procedure for the RTNEPH Infrared Water Vapor Attenuation Correction	68
6.1 Background	68
6.2 Radiance Calculations	69
6.2.1 Radiative Transfer Theory	69
6.2.2 Fast Frequency Integration	73
6.2.3 Transmittance Modeling	73
6.2.3.1 Water Vapor Lines	72
6.2.3.2 Water Vapor Continuum	76
6.2.4 Comparisons with Line-by-Line Calculations	78
6.3 RTNEPH Implementation and Timing Estimates	78
6.4 Evaluation of the New Water Vapor Attenuation Correction Procedure	84
6.4.1 The AFGWC Look-up Tables	84
6.4.2 Comparison Between the Current AFGWC Model and the New GL Model	85
6.5 Summary and Conclusions	102
7. AIMS/AMPS Development	105
7.1 System Description	105
7.2 System Expansion	107
7.3 System Capabilities	108
7.4 Summary	109
8. References	111
Appendix A	A-1
Appendix B	B-1
Appendix C	C-1
Appendix D	D-1
Appendix E	E-1
Appendix F	F-1

LIST OF FIGURES

Figure		Page
2.1	(a1) Horizontal cross section of vertical velocity at grid slab K=3; (a2) same as (a1) except K=4 (a3) same as (a1) except K=5; (a4) same as (a1) except K=6 (a5) same as (a1) except K=7 (b1) Vertical cross section of vertical velocity at grid slab I=8; (b2) same as (b1) except I=9 (b3) same as (b1) except I=10; (b4) same as (b1) except I=11 (c1) Vertical cross section of vertical velocity at grid slab J=9; (c2) same as (c1) except J=10 (c3) same as (c1) except J=11; (c4) same as (c1) except J=12	16 17 18 19 20 21 22
2.2	(a) Horizontal cross section of vertical velocity for case 1 at 1 km above the surface for model time = 1 hour; (b) same as (a) except model time = 1.5 hours. (c) same as (a) except model time = 2.0 hours	25 26
2.3	(a) Horizontal cross section of vertical velocity for case 2 at 1 km above the surface for model time = 1 hour; (b) same as (a) except model time = 1.5 hours (c) same as (a) except model time = 2.0 hours	27 28
5.1	The AFGWC RTNEPH grid system. The grid is superimposed on a polarstereographic map projection of the Northern Hemisphere	44
5.2	Schematic of consecutive satellite orbits depicting duplicate scan coverage. This figure is not to scale.	47
5.3	Correspondence between a Neph grid and satellite data. No data would be mapped to the circled grid points if each satellite data point was mapped to the nearest grid point. Instead, the nearest satellite data point is mapped to each grid point.	49
5.4	The relationship between azimuth angle of the satellite (α_s), the satellite subpoint latitude (ϕ_s), and the viewed point latitude (λ_s), and longitude (λ). Primes indicate variables when scanning in the negative direction. The azimuth angle is measured through 360 degrees eastward from north.	53
5.5	Binary Search Method. The large rectangle corresponds to a data array such that the data are oriented ascending/North up. The arrows mark the data latitudes and longitudes found by the search routine when it searches the first and last rows and columns. The small box in the center of the data array is the reduced array.	54
5.6	Schematic of AVHRR data is scanline format superimposed on a Northern Hemisphere Map projection. The RTNEPH grid is also displayed.	59

LIST OF FIGURES

(continued)

Figure		Page
5.7a	Examples of scan lines which cross only the dateline. The longitudes are converted to a + or - 360 deg base depending on which Neph box is being processed. Which base is designated by the (+) or (-) after the neph box number.	61
5.7b	Example of scan lines which cross both the dateline and the pole. The data set is divided at the scan line nearest the pole. The longitudes of lines crossing the dateline are converted to a + or - 360 deg base as in Figure 5.7a. The other scan lines are left unchanged.	62
5.8	Switch in latitude trend within a scan line due to curvature of the Earth. To the left of point A, latitude is monotonically increasing along the scan line. To the right, latitude is monotonically decreasing.	64
5.9	Division of data set into subarrays. The data are monotonic with respect to latitude within subarrays 1-5. The binary search technique is applied. The Random Search Box contains all non-monotonic data.	65
6.1	Relative spectral response of the OLS-T sensor.	71
6.2	Comparison of water vapor absorption profile in the 760-1030 cm^{-1} region obtained with Fascode (—) and the fast algorithm (---) for a standard tropical atmosphere and a 60 degree zenith angle. Continuum absorption is not included in this case.	77
6.3	Water vapor attenuation profiles in the DMSP/OLS-T band obtained with Fascode for different atmospheres and different values of the viewing angle (a). The corresponding differences between approximate calculations and Fascode results are plotted in (b).	79
6.4	Same as Figure 6.3	80
6.5	Same as Figure 6.3	81
6.6	Differences between the profiles of atmospheric attenuation obtained with the original version and the RTNEPH version of the fast radiative transfer algorithm (see text), for a standard tropical atmosphere and for a scan angle of 45°. In 6a the surface temperature is equal to the ambient air temperature at all levels. In 6b, it is equal to the ambient air temperature +5K.	83

LIST OF FIGURES

(continued)

Figure		Page
6.7	Comparison of the cloud analysis produced by the AFGWC and the GL models for Box #43. The OLS-T image (a) is a mosaic of three different satellite passes at approximately 6Z, 8Z, and 13Z on May 5, 1990. The AFGWC and GL cloud/no cloud decisions are shown in 7-b and 7-c respectively. Figures 7-d and 7-e show the fields of brightness temperature correction produced by the GL model and the AFGWC model. The difference between those two fields is plotted in 7-f. The satellite viewing angles are shown in 7-g. The times of satellite passages are indicated on this figure. As it is apparent in Figure 7-e, AFGWC tropical correction coincide with the 28°N latitude circle.	87
6.8	Same as Figure 7 for Box #52. As indicated in Figure 8g, the OLS-T data correspond to two satellite passes at approximately 12Z and 13Z, on May 5, 1990.	90
6.9	Scatter plots of the differences between modeled and measured OLS-T brightness temperatures over clear land areas in Eastern Europe (Neph Box #30) versus secant of the satellite viewing angle. This figure shows the errors produced when (a) the modeled skin surface temperatures are uncorrected, and when the AFGWC correction (b) and the GL correction (c) are applied. The data correspond to two satellite orbits at approximately 7Z and 9Z on May 5, 1990.	95
6.10	Same as Figure 6.9 but for Eastern Sahara (Box #31). The data correspond to two satellite orbits at ~7Z and 9Z.	96
6.11	Same as Figure 6.9 but for Southeastern U.S. and Southern Mexico (Box #52). The data correspond to one satellite orbit at ~12Z.	97
6.12	Same as Figure 6.9 but for North Atlantic ocean off the coast of Europe, west of 10°W (Box #38). The data correspond to one satellite orbit at ~11Z	98
6.13	Same as Figure 6.9 but for North Atlantic ocean off the coast of Europe, west of 10°W (Box #38). The data correspond to one satellite orbit at ~11Z.	99
6.14	Same as Figure 6.9 but for Atlantic ocean and part of the Caribbean Sea (Box #53). Data correspond to three satellite orbits between ~10Z and 13Z.	100
6.15	Same as Figure 6.9 but for Atlantic ocean (Box #54). Data correspond to one satellite orbit at ~12Z.	101
7.1	AIMS/AMPS system configuration.	106

LIST OF TABLES

Table	Page
3-1 NDB data dictionary structure	32
4-1 Mapping of 6-bit UNISYS data to VAX data structure	37
4-2 Satellite ephemeris data structure	38
4-3 AFGWC upper air data packing format by 36-bit word number	38
4-4 RDNEPH upper air data structure	39
4-5 AFGWC surface temperature data packing format by 36-bit word	40
4-6 AFGWC background brightness data packing format by 36-bit word	41
4-7 AFGWC terrain and geography data packing format by 36-bit word	42
6-1 Polynomial coefficients for spectral lines of water vapor for the nine 30 cm ⁻¹ subintervals covering the 760-1030 cm ⁻¹ region	75
6-2 Coefficients for self and foreign broadened water vapor continuum for the nine 30 cm ⁻¹ subintervals covering the 760-1030 cm ⁻¹ region	76
6-3 Summary of look-up table dependence on physical parameters	84

1. Introduction

This final report describes work that was performed at the Geophysics Laboratory (GL) under a research support contract with the Atmospheric Prediction (LYP) and Satellite Meteorology (LYS) groups within the Atmospheric Sciences Division (LY). Three tasks were identified under this contract, these were:

- 1) perform and evaluate Advance Meteorological Processing (AMPS) studies;
 - 2) devise, develop, improve and provide satellite cloud and precipitation techniques;
- and
- 3) provide on-site technical support to maintain and upgrade the satellite receiving station and associated processing and display equipment for the Air Force Interactive Meteorological System (AIMS) and AMPS.

AMPS studies were in two parts. The first involved development of a short term forecasting algorithm for prediction of storm movement based on trend extrapolation of either radar or satellite data and is described elsewhere (Bianco and Huang, 1990; Heideman et al., 1990). The second part was an investigation of a mesoscale prediction model that was developed for GL by the Aster Corporation. A key feature of the model is the ability to assimilate radar velocity data to adjust the model wind field toward the observed values. Initial evaluations of the model by the Aster Corporation were disappointing. AER performed a detailed analysis of the model based on a literature search, analysis of the program source code, and test runs on case study data. Details of the investigation and modifications made to the model user environment are described in Section 2.

Work in the area of satellite cloud and precipitation techniques centered around the Air Force operational cloud analysis model, RTNEPH. GL maintains a research and development version of the RTNEPH satellite processor known as RDNEPH which has long served as a testbed for new and innovative research related to RTNEPH. AER provided support in a number of areas related to the nephanalysis effort including: development of interactive cloud analysis validation techniques (Gustafson and Felde, 1989), evaluation of a new cloud layer clustering algorithm (d'Entremont et al., 1989), and an investigation of the effect of improved terrain height information on nephanalysis accuracy (Schaaf et al., 1989). The RDNEPH effort is largely dependent on AFGWC as a data source through access to the hemispheric RTNEPH data bases. AER developed a data base management capability for the RDNEPH project to handle the large volume of case study data received from AFGWC. This capability was implemented on the dedicated LY computer system known as the Air Force Interactive Meteorological System (AIMS). A

complementary facility was also developed to decode AFGWC data tapes and to place the tape data online in the AIMS data base. These programs are described in Sections 3 and 4 respectively. Satellite data obtained from AFGWC are normally acquired from the two channel DMSP OLS sensor. However, much of the recent work at GL has been on multispectral cloud analysis using the five channel NOAA AVHRR instrument. AVHRR data are generally available from NOAA in a standard format. GL had on hand a program to decode the five channel tape data and to store it on AIMS in the satellite scan line format. However, this format is incompatible with the RDNEPH program which requires that all data be mapped to the AFGWC standard grid based on a polar map projection. AER developed a program to transform the five channel AVHRR data from scan line projection to the AFGWC polar stereographic projection at the required grid resolution. The image transformation program is detailed in Section 5.

A major research effort was put forth to develop a procedure to predict atmospheric attenuation effects on infrared radiation measured by an IR radiometer on a Low Earth Orbiting (LEO) platform. This is a key aspect to the RTNEPH infrared processor which performs a cloud/no cloud decision based on the difference between predicted clear scene temperatures and satellite observed infrared brightness temperatures. In the RTNEPH, atmospheric effects on IR radiation are modeled through a series of empirical look-up tables. A new technique was developed to predict atmospheric transmission based on a radiative transfer band model calculation. The band model was developed, implemented in the RDNEPH, and tested on case study data obtained from AFGWC. Results of this effort are in Section 6.

The final task, to provide technical support to the dedicated LY research computer system is reported in Section 7. AER provided both hardware and software support on a day-to-day basis and worked to enhance the system capabilities through long term system and application software development.

2. Investigation of the GL Mesoscale Forecast Model

Observations of radial wind velocity obtained from Doppler radar are considered to be useful in improving short-term regional forecast models. Recently, Cotton et al. (1985) implemented a technique which uses frequent radar updates to nudge the modeled horizontal wind field toward the observed radar wind field. The Cotton work was performed using the Colorado State University (CSU) Regional Atmospheric Modeling System (RAMS). As a result of those efforts two simplified versions of the RAMS model, suitable for implementation on micro computers, were delivered to the Geophysics Laboratory. Initial tests of these models were disappointing, in simulations performed by Cotton the nudging scheme failed to improve the forecast in both versions. It was concluded that it was insufficient to modify the modeled wind field without also changing the thermodynamic field, when this is the case the model tends to restore itself back to the original state in which no nudging occurred. Similar results based on more sensitive tests were also reported by Yu-Chieng Liou (1989). AER performed an investigation of both versions of the RAMS model with two goals: 1) to obtain a better understanding of how the models operate and provide more complete documentation and 2) to suggest possible modifications to the models to improve their performance. In conjunction with the analysis a number of modifications were made to the run-time environment at GL to simplify model operation and improve interpretation of results.

The AER work focused on three major areas: 1) to analyze the model characteristics through a literature review and analysis of the source code; 2) to develop an understanding of the model run-time environment through investigation of the required preprocessing program and run-time options; and 3) to improve display capabilities of the model results. The following three subsections address each of these areas in turn. The model was installed on the GL Advanced Meteorological Processing System (AMPS) and test runs on case study data were performed to further study the model characteristics and attempt to duplicate the results reported in Cotton et al. (1985). The conclusions from that effort are described in Section 2.4 followed by a summary in Section 2.5. Related information is contained in Appendices A through C.

2.1 Model Characteristics

Two versions of the modified RAMS model were delivered to GL by the Aster Corporation. One is a hydrostatic model with a horizontal domain of 100 by 100 km and a 5 km resolution. The complete set of equations includes prognostic equations for the u

and v wind components, potential temperature, and vapor mixing ratio (as a passive tracer only) and diagnostic equations for vertical velocity and pressure. The vertical domain of the atmosphere contains 8 layers and the depths range from 400m to 750m, 4 levels are used in the soil model. Some simplified numerical schemes were used in the model. There is a second-order advection scheme, a constant pressure top boundary condition for the hydrostatic equation and a radiative lateral boundary condition with a specified phase velocity. The physical parameterizations include a simplified diffusion scheme (e.g., only use the vertical shear to compute the deformation), a modified radiation scheme, and a modified convective parameterization, the latter two use information from an input sounding. The second version is a further simplification and is called the Mix-Layer model. It uses the same horizontal computational domain as the hydrostatic model. It contains a superadiabatic surface layer with a depth of 50m, a neutral mixed layer above having constant vertical profiles of momentum, potential temperature, and water mixing ratio, a possible inversion of zero vertical thickness capping the mixed layer and a stable layer above them all. The assumed vertical structure limits the model's ability in simulating vertical motion. The Mix-Layer model was reported to produce a forecast in 1/6 real time on a Micro VAX II class computer system while the full hydrostatic version achieved 1/5 of real time in simulation on the same class of machine. For further detail about the models, please refer to Cotton et al. (1988).

Early in the investigation a decision was made to concentrate on the full hydrostatic version of the model. This decision was based on the assumption that the physical schemes and assumptions of the Mix-Layer model were overly simplified and that further work was not warranted. Hereafter, the hydrostatic version will be referred to as the GL-3D model.

The GL-3D model supports a large number of run-time options which control various analysis and control parameters. These options are specified through a FORTRAN namelist structure contained in a job control file. They include parameters to define the analysis resolution, model time scales, number of vertical layers, geographical position, output parameters, etc. A more complete description of the run-time options is provided in Section 2.2 and Appendix A.

The time integration scheme implemented in the GL-3D model is called a time-split scheme. Since all aspects of the model forecast procedure are not yet fully understood, a review of the time-split scheme from the literature was conducted. A simple example of this scheme can be found in Tremback et al. (1985). Time-split scheme is a technique for applying different time steps for the time integration scheme to properly handle fast propagating features (e.g., Lamb wave or external gravity waves in a hydrostatic system).

The time-split scheme computes the terms responsible for the propagation of the fast mode on a smaller timestep, and runs simultaneously with slower moving modes on a longer timestep. With the knowledge gained so far from the analysis of the model and from the Tremback paper, the following equations which include the prognostic equations of u , v , and θ , the continuity and hydrostatic equations, and the boundary conditions are used to demonstrate the forecast procedure of the time-split scheme of the GL-3D model;

$$\begin{aligned} \frac{\partial u}{\partial t} + \theta \frac{\partial \pi}{\partial x} = & -u \frac{\partial u}{\partial x} - v \frac{\partial u}{\partial y} - w \frac{\partial u}{\partial z} + fv + \frac{\partial}{\partial x} \left[K_{mx} \frac{\partial u}{\partial x} \right] \\ & + \frac{\partial}{\partial y} \left[K_{my} \frac{\partial u}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_{mz} \frac{\partial u}{\partial z} \right] = F \end{aligned} \quad (1)$$

$$\begin{aligned} \frac{\partial v}{\partial t} + \theta \frac{\partial \pi}{\partial y} = & -u \frac{\partial v}{\partial x} - v \frac{\partial v}{\partial y} - w \frac{\partial v}{\partial z} - fu + \frac{\partial}{\partial x} \left[K_{nx} \frac{\partial v}{\partial x} \right] \\ & + \frac{\partial}{\partial y} \left[K_{ny} \frac{\partial v}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_{nz} \frac{\partial v}{\partial z} \right] = F_v \end{aligned} \quad (2)$$

$$\begin{aligned} \frac{\partial \theta}{\partial t} + w \frac{\partial \theta}{\partial z} = & -u \frac{\partial \theta}{\partial x} - v \frac{\partial \theta}{\partial y} + \frac{\partial}{\partial x} \left[K_{hx} \frac{\partial \theta}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_{hy} \frac{\partial \theta}{\partial y} \right] \\ & + \frac{\partial}{\partial z} \left[K_{hz} \frac{\partial \theta}{\partial z} \right] + S_\theta = F_\theta \end{aligned} \quad (3)$$

$$\frac{\partial \rho w}{\partial z} = -\frac{\partial \rho u}{\partial x} - \frac{\partial \rho v}{\partial y} \quad (4)$$

$$\frac{\partial \pi}{\partial z} = -\frac{g}{\theta} \quad (5)$$

where

$$\pi = c_p \left[\frac{p}{p_o} \right]^{R/c_p} \quad (6)$$

where R is the gas constant for dry air, c_p is the specific heat of dry air at constant pressure, p is pressure and p_o is a constant. Those terms with coefficients K_m , K_n , and K_h in (1), (2) and (3) are diffusion terms.

The pressure at $z = 0$ is determined from a prognostic equation derived from substitution of the hydrostatic equation into the compressible continuity equation, and with $w=0$ at the surface.

$$\frac{\partial p}{\partial t} = -g \int_0^{Z_t} \left(\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} \right) dz \quad (7)$$

where Z_t is the top of the computational domain.

The forward-backward time differencing scheme is as follows:

- (a) The right hand side of (1), (2) and (3), F_u , F_v and F_θ are computed,
- (b) θ is stepped forward to $t + \Delta t_L$ where Δt_L is the long timestep,

$$\theta^{*t+\Delta t_L} = \theta^t + F_\theta \cdot \Delta t_L$$

- (c) θ is stepped forward to $t + \Delta t_S$ where Δt_S is the small timestep,

$$\theta^{t+\Delta t_S} = \theta^{*t+\Delta t_L} - \left[w^t \frac{\partial \theta^{*t+\Delta t_L}}{\partial z} \right] \Delta t_S$$

- (d) Pressure at $t + \Delta t_S$ is computed with (5),
- (e) The horizontal velocity is stepped to $t + \Delta t_S$,

$$u^{t+\Delta t_S} = u^t - \left[\theta \frac{\partial \pi^{t+\Delta t_S}}{\partial x} - F_u \right] \Delta t_S$$

$$v^{t+\Delta t_S} = v^t - \left[\theta \frac{\partial \pi^{t+\Delta t_S}}{\partial y} - F_v \right] \Delta t_S$$

(f) The vertical velocity at $t + \Delta t_S$ is computed with (4),

(g) The pressure boundary condition is updated to $t + \Delta t_S$ with (7) using the divergence at the $t + \Delta t_S$ level,

(h) The small timestep, (c)-(g), is repeated n times until $n\Delta t_S = \Delta t_L$

When the time integration is performed, the following boundary conditions are applied in the GL-3D model to assure proper spatial numerical integration:

At the eastern and western boundary:

$$\frac{\partial v}{\partial x} = 0, \quad \frac{\partial \theta}{\partial x} = 0,$$

and at the eastern boundary:

$$\frac{\partial u}{\partial x} = 0$$

At the northern and southern boundary:

$$\frac{\partial u}{\partial y} = 0, \quad \frac{\partial \theta}{\partial y} = 0,$$

and at the northern boundary:

$$\frac{\partial v}{\partial y} = 0$$

At the top of the domain, the first derivative in the vertical direction equals that at the level just below the top of the domain;

$$\left(\frac{\partial u}{\partial z} \right)_{nz} = \left(\frac{\partial u}{\partial z} \right)_{nz-1}, \quad \left(\frac{\partial v}{\partial z} \right)_{nz} = \left(\frac{\partial v}{\partial z} \right)_{nz-1}$$

$$\left(\frac{\partial \theta}{\partial z}\right)_{nz} = \left(\frac{\partial \theta}{\partial z}\right)_{nz-1}$$

and

$$\left(\frac{\partial w}{\partial z}\right)_{nz} = 0$$

Therefore, the values at the upper boundary can be derived from the information below the upper boundary; e.g.,

$$U_{nz} = U_{nz-1} + \frac{Z_{nz} - Z_{nz-1}}{Z_{nz-1} - Z_{nz-2}} (U_{nz-1} - U_{nz-2})$$

where nz indicates the upper boundary, $nz - 1$ is the level just below the upper boundary, and so on.

At the bottom of the domain:

$$\frac{\partial u}{\partial z} = 0, \frac{\partial v}{\partial z} = 0, \frac{\partial \theta}{\partial z} = 0,$$

The advantage of the time-split scheme is that it allows a longer time step to be used in the model thereby saving computational time.

2.2 Model Run-Time Environment

When the GL-3D model was delivered to GL it retained many of the idiosyncrasies of the computer environment it was developed in. An in depth understanding of the program's maintenance and configuration was required before it could be effectively used at GL. For instance, if a change were to be made in one of the program modules, it was necessary to understand and run a complex command procedure to create a new executable code. All source code was written in a non-standard programming language that required a preprocessor program to convert it to standard FORTRAN. The preprocessor language was apparently developed at CSU and is constantly undergoing modifications and updates. Since no source code for the preprocessor program was provided, it was necessary to replace the CSU preprocessor to avoid dependence on a "black box" program and to allow

for future modifications to the model. Since the GL-3D model is to be used as a research tool in the future, the ease of its use and the transparency of its modules needed to be maintained from the start. During the AER investigation, the GL-3D model has undergone several changes designed to make it more straight forward to use and manipulate.

2.2.1 Preprocessor

To eliminate the CSU preprocessor a translator was created which reads the original program modules, modifies the code based on an analysis of the CSU preprocessor, and writes the modified code as standard FORTRAN into new modules. However, before the translator could be tested the GL-3D modules had to be made VAX/VMS specific. The original model was set up to run on any of four machines, the CSU preprocessor customized the code for the specified machine. The machine choices are: Cray XMP/COS, Cyber 205, VAX/VMS, and Cray XMP/UNICOS; principal differences center around the input/output capabilities of each machine. All machine dependent I/O code has been removed from the source files in favor of VAX/VMS code. The line numbers for machine specific changes have been saved in a file for reference if another machine is used in the future.

In addition to the I/O considerations, four other conditions are handled by the CSU preprocessor that support user selectable options. These options are: latitude/longitude grid spacing, no latitude/longitude grid spacing, topography, and no topography. The model was written such that the actual lines of code that implemented these options are either enabled or ignored by the preprocessor depending on a key letter placed at the beginning of the line. If latitude and longitude grid spacing is used, the line of the code is preceded by an 'L', whether topography is used or not is denoted by 'Z' or 'Y' respectively. These options were left in the code because they directly affect the way the code is optimized by a vectorizing compiler. Whenever an new option is specified, the code must be re-vectorized by the compiler. Therefore, the new translator passes all of these lines through so that a new simple preprocessor is still necessary before the code can be compiled. While replacing the old preprocessor with a new one may seem contradictory it was felt that this was the simplest way to retain the selected options without affecting the optimal vectorization of the code. The advantages of this new system are: 1) once the new translator is run, standard FORTRAN code is created which is easily understood and modifiable; 2) it is only necessary to run the translator once, from then on all work is performed on the FORTRAN code; and 3) the new simple preprocessor was created at GL, is understood, and is under the control of the GL staff.

The process of validating the accuracy of the translator was set up as a feed-back loop. A module would be sent through the translator and its output saved. A copy would be sent to the CSU preprocessor and this output would be compared with the translator output. An unexpected difference meant that the translator had to be adjusted and the file returned to the updated translator for another try. This continued until all the GL-3D modules passed through the loop with no unexpected differences occurring. An expected difference between the output from the translator and the CSU preprocessor was one of the four conditions described above. To prevent the CSU preprocessor from deleting some of these conditions and thereby disrupting the comparison, each condition was activated in the CSU preprocessor before the file was sent to it. This kept all the lines in the file intact, but removed the CSU preprocessor activation character as a prefix to those lines that had it. Since the new translator left the activation characters associated with the four conditions intact, a direct comparison found differences in the two files, however, such differences were expected and ignored during the comparison.

Once the translator was determined to be correct for all the expected changes made by CSU preprocessor, the new preprocessor (AER preprocessor) was created. Files leaving the translator retained only three valid activation characters: Z or Y for topography or no topography and L or no L for latitude-and-longitude spacing or no latitude-and-longitude spacing respectively. An example of the original code and the corresponding logical 'if-then-else' operations follows:

<u>ORIGINAL</u>		<u>IF-THEN-ELSE STATEMENT</u>
	X = A +	IF (Z AND L) THEN X = A + B + D + E
Z	B +	ELSE IF (Z AND NO L) THEN X = A + B + E
Y	C +	ELSE IF (Y AND L) THEN X = A + C + D + E
L	D +	ELSE X = A + C + E
	E	END IF

Note that the AER preprocessor does not replace the lines on the left with an actual FORTRAN if-then-else statement because this would cause the vectorizing capability to be lost. Rather the original source code lines developed by the Aster Corp. are retained.

Include files are used extensively in the GL-3D model to maintain global variables throughout the program execution. These variables appear as common blocks and namelists. In order to properly dimension the arrays in the common blocks, parameter statements are used to specify the dimension sizes. In the original code the parameter declarations were maintained separately from the common block definitions in different

include files. To avoid confusion, all of these dual declaration and definition include files were combined into two include files called STORAGE.INC and CONSTOR.INC which are required by all the routines in the GL-3D model. A program called MK_INCLUDE was created to automate the include file creation process via an interactive response format.

The AER preprocessor also uses the include files to determine which of the four user options to enable. In the parameter lists contained in STORAGE.INC, two flag parameters called LATLON and ITOPO are declared. The preprocessor scans for these flags and depending on their values sets the appropriate user options:

```
LATLON= 1   nclude latitude-and-longitude,  
LATLON= 0 : Suppress latitude-and-longitude,  
ITOP0  = 1 : Include topography,  
ITOP0  = 0 : Suppress topography.
```

It is important to note that the program MK_INCLUDE should be run before the AER preprocessor is run in order to create the proper FORTRAN code to implement the selected options.

Model results obtained using the translator and new preprocessor were tested against results obtained using the CSU preprocessor to insure that the modifications had not affected the model output. Both versions were run on the same case data with the same selected conditions and the same input job control file. The graphical output was checked visually and no difference was found between the two versions.

2.2.2 Job Control File

The GL-3D model is run by reading in the values of the control parameters in a job control file (see Appendix A). Some information on the use of these parameters was provided when the model was delivered to GL. However, during the model analysis it was discovered that this information was incomplete, and in some cases incorrect. In the following section some guidelines are provided on the usage of certain key parameters.

Prior to execution of the program, the basic configuration of the model should be fixed. The number of horizontal grid points (NNXP, NNYP) and the horizontal grid size (DELTAX, DELTAY) should be matched with the format of topographic data (SURFILE). The latitude of the southern edge (RLAT) and the longitude of the western edge (WLON1) of the computational domain should also be consistent with input topographic data. Correct values of the latitude and longitude are necessary to ensure proper calculation of

input solar flux and Coriolis force. The height of the vertical levels in the model are specified by the number of levels and the vertical spacing (NNZP, DELTAZ, DZRAT, DZMAX). For example, if NNZP = 10., DELTAZ=400., DZRAT=1.5, and DZMAX = 750, then:

Level 1	0 m,
Level 2	400 m,
Level 3	1000 m,
Level 4	1750 m,
Level 5	2500 m,
Level 6	3250 m,
Level 7	4000 m,
Level 8	4750 m,
Level 9	5500 m,
Level 10	6250 m.

The radiation parameterization (NIRAD) and soil model (NISFCL) are required.

Case study data from the 1987 CINDE field program were provided with the model by the Aster Corp. Required input data are surface observations, rawinsonde profiles, radar velocity data and, optionally, topographic heights. Data formats are in ASCII. Once a case study data set is selected, the corresponding input data file names are defined in the job control file (SURFILE, SNDFILE, NUDFILE). The local solar time at beginning of simulation (STRTIM) and the date (IMONTH1, IDATE1, IYEAR1) should be fixed according to the time of the input data files. The format of the input sounding profile file is fixed in the GL-3D program. Although there is a section in the job control file for specification of a new sounding, these values are ignored by the model. It would require a relatively simple modification to the program to accept these values, however, this effort was not a part of this contract.

The user can control how long the simulation should be run (TIMMAX), how often the history files will be updated (IOUTPUT, FRQHIS, MAXHFL, NHWDMX, HFILOUT(1), HFILOUT(2)), and how often the model will print out or plot the field data (FRQPRT). The simulation can be started at time zero or it can be restarted with a history file at the time associated with the history file (IFILSTR, TIMSTR, HFILIN).

The user can control how the model attempts to nudge the wind field toward the observed radar wind. With the choice of different types of nudging (NUDRAD) and the time to start and end the nudging (STRTNUD, ENDNUD), the user can access the effect of the nudging method. It is also possible to adjust the frequency of nudging (SCLNUD).

2.3 Display of Model Results

The original model included modules to produce graphical output of the analysis results. Supported output formats were either horizontal or vertical cross sections for selected output fields or a tabular listing of numeric values. The parameters in the \$PRNT section of the job control file determine which dynamic fields are to be output for examination (HPLT, IPLFLD) and what the output format will be (PLFMT(3), IPLTYP). If a cross sectional analysis is selected, other parameters in the \$PRNT section control the direction of the analysis (IXSCTN, ISBVAL), the region or subregion of the computational domain to be viewed (IAA, IAB, JOA, JOB), and whether a velocity wind vector is to be overlaid on the graphic output (IPLVECT). A set of sample parameter values for two different types of output are given in Appendix C.

2.3.1 Model Output

The GL-3D model calculates the values for 16 different forecast fields for all points in a user specified three dimensional grid at a specified model time frequency. In the input job file the user specifies which fields to plot, in which plane (XZ, YZ, or XY), and at which grid slabs. As output the GL-3D model stores a graphics metafile which can be displayed as a black and white contour analysis of the specified field. Contours are drawn and labeled at intervals chosen by the plotting routine, negative values are drawn with dashed lines. Wind vectors can be optionally overlaid on the analysis to characterize the horizontal or vertical wind field. To obtain plots for any field or grid cross-section not specified in the job file the entire GL-3D program must be rerun. In order to avoid this time-consuming procedure a method was developed to store calculated values for all fields at all points in the three dimensional grid into a data file. At a later time the values can be quickly read from the file and plotted interactively.

For consistency it was decided to store the data in a format that could be accessed by the existing display modules. These modules were then used to construct a separate program to extract the stored data and create output identical to the output from the model program itself. The model uses four main output subroutines (PRTWHAT, PRTOPT, SFCPRT, and PRTOUT). The code was modified so that each time one of these four subroutines is called the following information is written to the output data file:

1. the subroutine name,
2. the subroutine arguments,

3. all common block variables.

Included among the common block variables, if relevant to that subroutine, are the model time and the values of all 16 fields at all three dimensional grid points. These are all stored independent of the fields specified in the job control file.

To run the GL-3D model with data-storing capability, three changes are necessary:

1. Link the program with RDRIVAF_RAMSTOR and RPRNTAF_RAMSTOR, instead of RDRIVAF and RPRNTAF.
2. Add to the include file STORAGE.INC the statement
COMMON /STORE/ LUSAVE,
where LUSAVE is the logical unit assigned to the output data file.
3. At run time define VMS logical RAMSDAT to the data storage file.

2.3.2 Reading Stored Data

The separate routine created to extract the stored data is called VIEW_PLOT. VIEW_PLOT reads the data records from the data file, interprets the subroutine name, and passes the arguments and common block variables to that subroutine. The output is directed to a graphics metafile in exactly the same way as the forecast model. Time-dependent subroutines are only called by VIEW_PLOT if the stored model time agrees with the user specified input time.

User inputs are specified in a separate job control file that is made up of the 'PRNT' section of the model job file plus two additional fields: TIME_INPUT, the model time at which to create a plot, and ICOLOR_PLOT, which is 1 for color output and 0 for black and white (see Appendix B). Color capability was added using the subroutine CONPACK (NCAR Graphics Version 3). The contours are color-filled according to value.

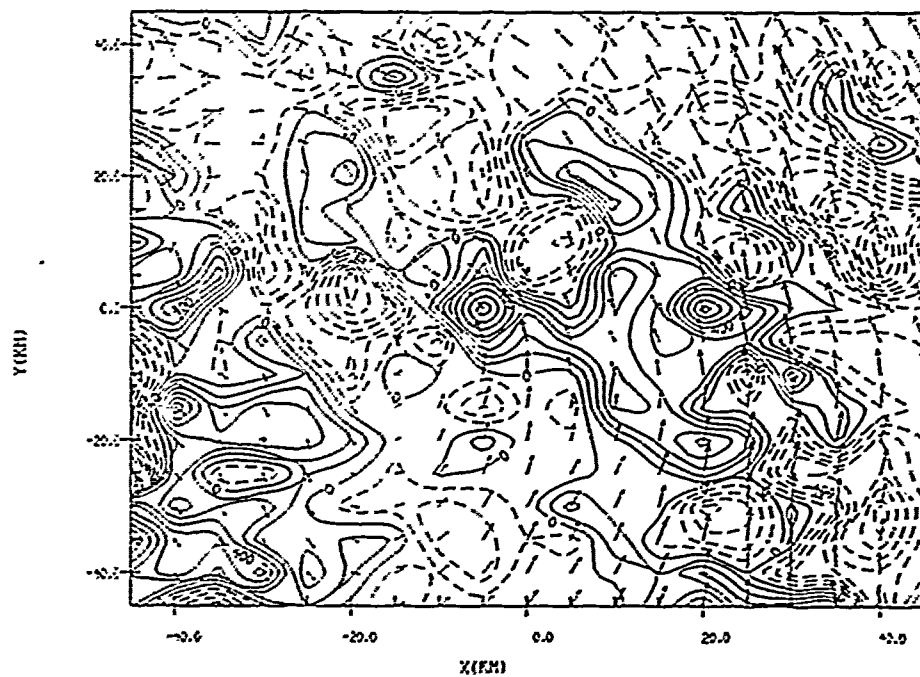
To run VIEW_PLOT, it is first necessary to define the VMS logical RAMSDAT as the input storage data file containing the full set of 3-D grid data. For black and white output link VIEW_PLOT with RPRNTAF_VP, V_P_SUBROUTINES, and RAMSGKS. For color output link with RAMSCON instead of RAMSGKS. The model output modules with which VIEW_PLOT is linked can also be run through the AER preprocessor in order to specify the topography or latitude/longitude grid spacing options.

Typically, the GL-3D model can produce a 2 hour forecast, storing data every 30 minutes of model time, in about 95 minutes of CPU time. VIEW_PLOT can create a metafile displaying a single field, at one input time, in just 13 seconds CPU time.

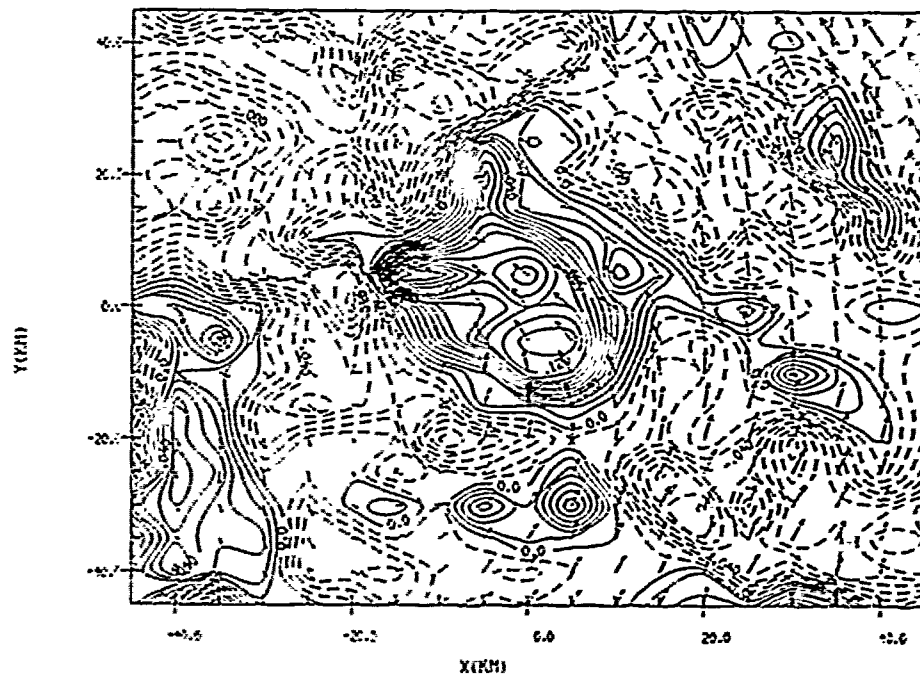
2.3.3 Graphical Display

An interactive program called VIEW_META was written to display the metafile output produced by either the GL-3D model or the VIEW_PLOT program. The user will be prompted to specify the name of the data file containing the metacode output. First the graphic output will be shown sequentially, then the user will be prompted to choose to view the new metafile, to review the plots sequentially, or to review any cross section specified. A new display program which incorporates the Itrans and Ctrans programs of the NCAR graphics package (version 3.0) in the DECwindow environment is under development. With the metafile output generated from the full 3-D data set, the prototype of this program allows the user to interactively specify the data field to be viewed and the grid slab for which the cross section is to be displayed. The user can select any combination of grid slabs in any order to be viewed sequentially. A separate version of the program called VIEW_META2 has been created to produce hard copy plots. Prior to executing VIEW_META2, the VMS logical GKSSWSTYPE must be defined as either 38 for the LN03 laser printer black and white output or 91 for the LJ250 color printer. VIEW_META2 creates a file which can be sent directly to the selected hard copy device. Using VIEW_META, output produced directly by the GL-3D model were compared to corresponding output produced by VIEW_PLOT and the results were identical.

An example of the output from VIEW_PLOT and VIEW_META2 is contained in Figure 2.1. Here a series of plots are used to provide a three dimensional representation of a feature of interest within the 3-D grid. Plotted are a series of vertical (XZ, YZ) and horizontal (XY) cross sections at different grid slabs for a one hour forecast of the vertical wind component (W). Vertical velocity has a maximum near the center of the plot around $X = -5$, $Y = 0$. The data grid has 21 points in the x and y directions and 10 in the z direction, the slabs chosen for Figure 2.1 are I (or X) = 8 to 11, J (or Y) = 9 to 12, and K (or Z) = 3 to 7. Contours are drawn at intervals of .01 cm/sec, contour labels are scaled by a factor of .1E+05. Appendix B contains the VIEW_PLOT job file used to produce this figure.

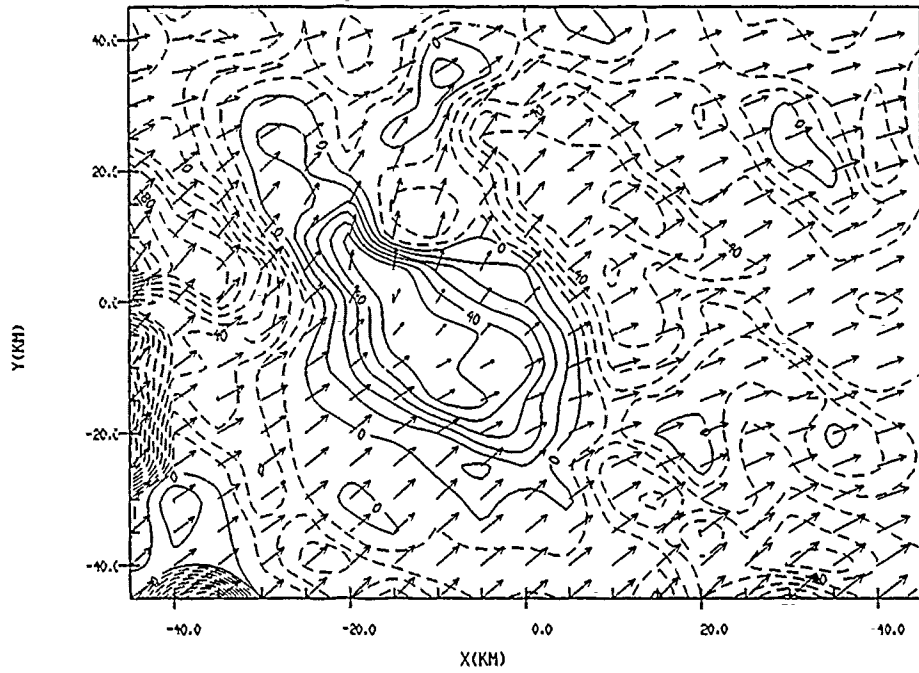


a1

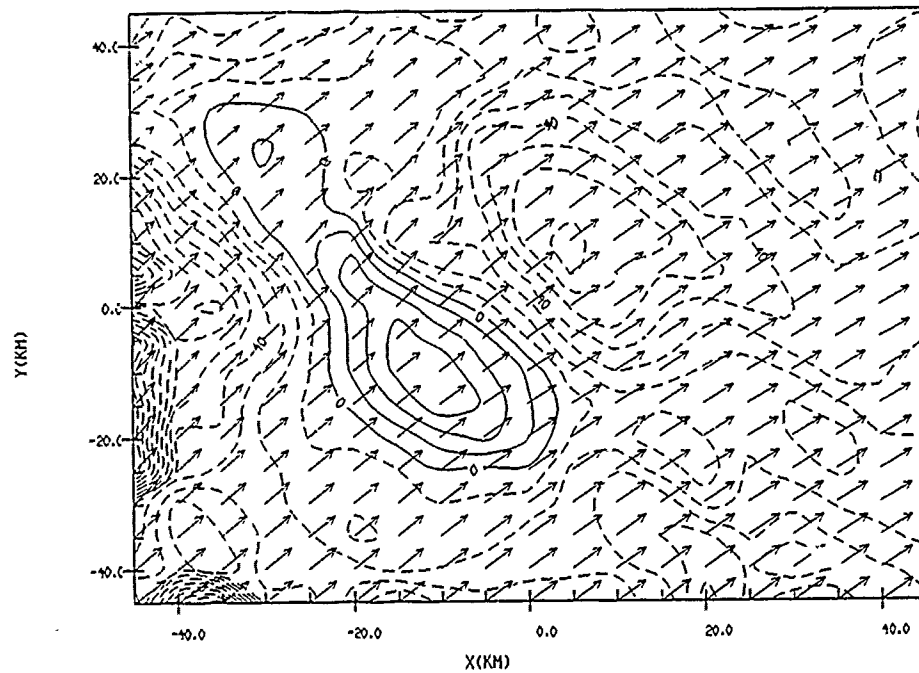


a2

Figure 2.1 (a1) Horizontal cross section of vertical velocity at grid slab $K=3$; (a2) same as (a1) except $K=4$.



a3



a4

Figure 2.1 (continued) (a3) same as (a1) except $K=5$; (a4) same as (a1) except $K=6$.

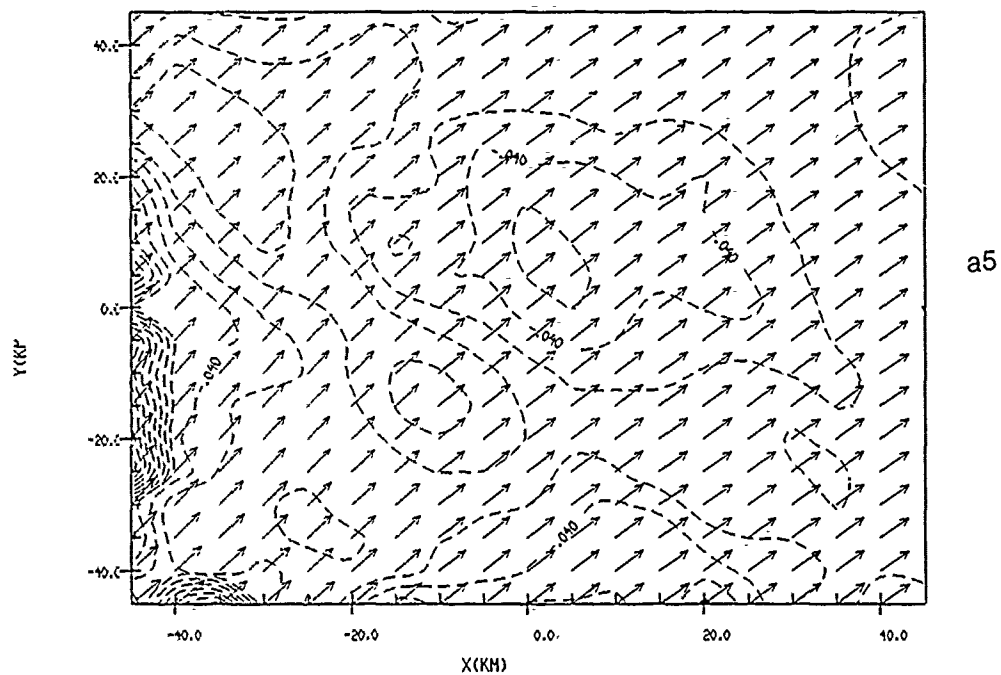
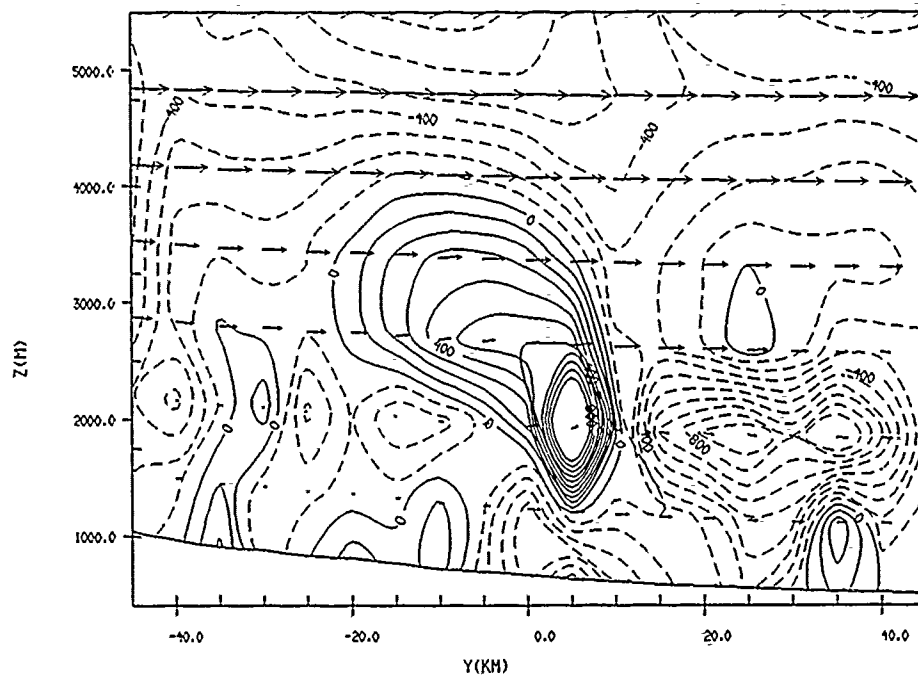
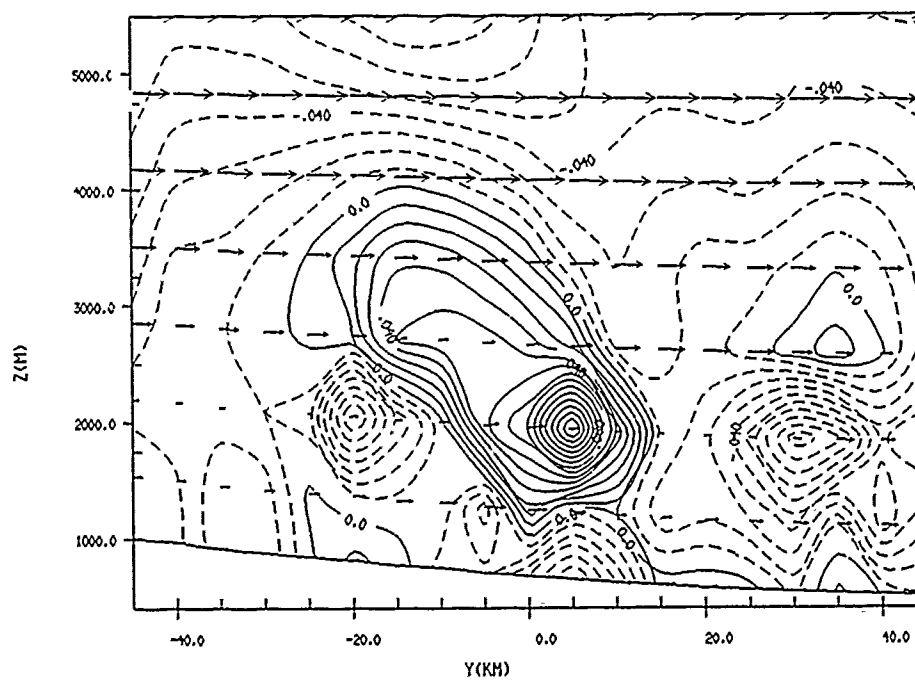


Figure 2.1 (continued) (a5) same as (a1) except $K=7$.

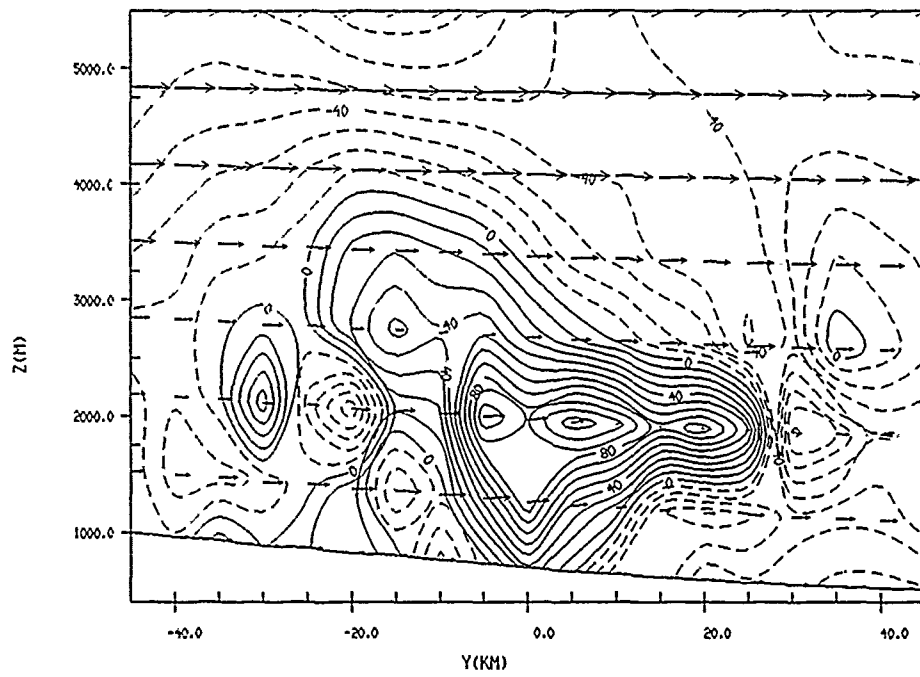


b1

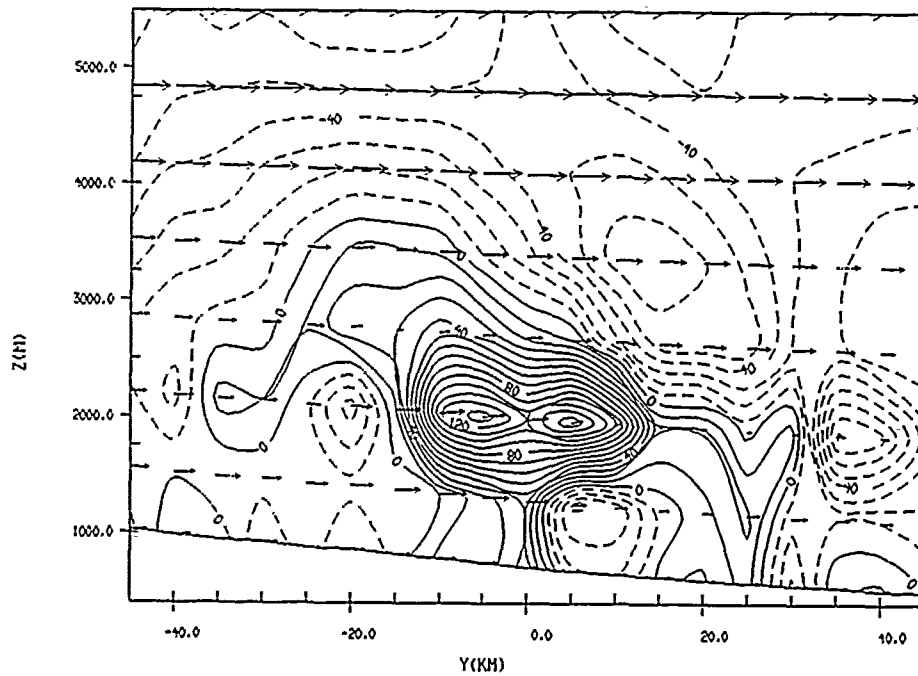


b2

Figure 2.1 (b1) Vertical cross section of vertical velocity at grid slab I=8; (b2) same as (b1) except I=9.

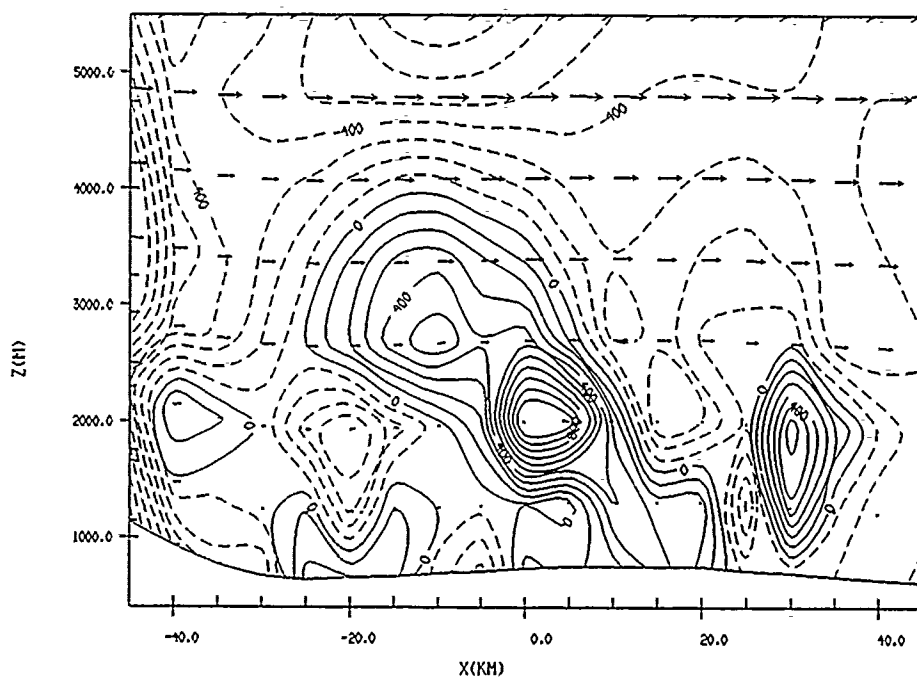


b3

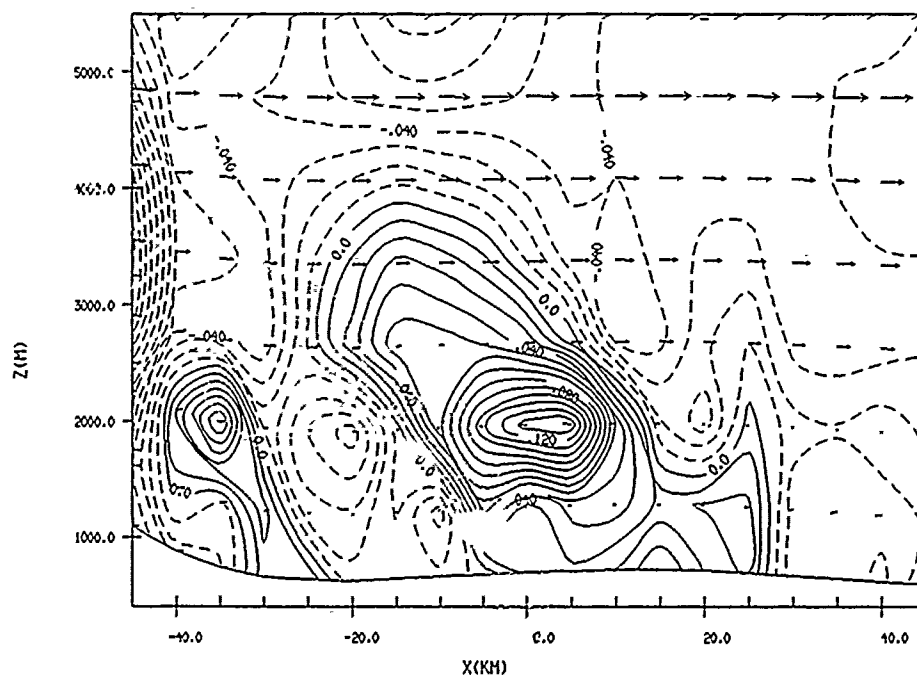


b4

Figure 2.1 (continued) (b3) same as (b1) except $I=10$; (b4) same as (b1) except $I=11$.

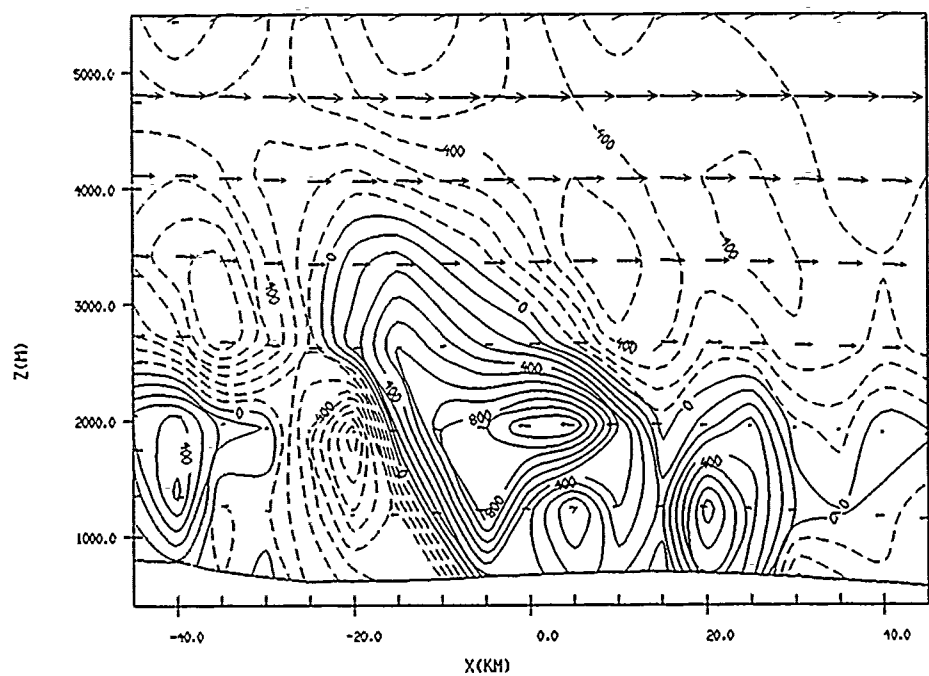


c1

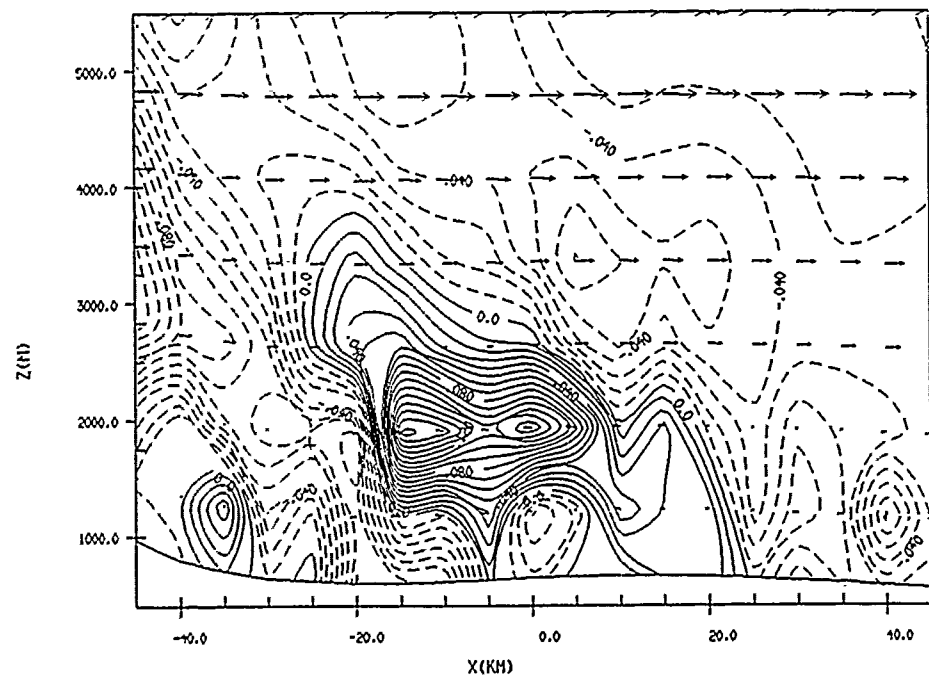


c2

Figure 2.1 (c1) Vertical cross section of vertical velocity at grid slab J=9; (c2) same as (c1) except J=10.



c3



c4

Figure 2.1 (continued) (c3) same as (c1) except J=11; (c4) same as (c1) except J=12.

2.4 Comparison Results

Following the analysis of the model source code and related documentation, attempts were made to reproduce the model results reported in Cotton et al. (1988). The only points of comparison available were the figures contained in the Cotton report. The CINDE case data delivered with the model were assumed to be the same as what was used to test the model prior to delivery. This was to be the final check that the changes that had been made to the program had not resulted in any modification to the processing logic. However, during this process several values in the job control file were found to be inconsistent with the conditions specified in the report.

Initial comparisons were made between results from two versions of the GL-3D model, one produced using the CSU preprocessor and the other using the combined translator and AER preprocessor. Identical results were obtained from both of these programs when run using the same job control file. A comparison was then made between the GL-3D model results and the results reported in the Cotton report using the job control file delivered with the model. Very different results were obtained from what was in the report, although the two versions of the GL-3D model again produced identical results. Based on the latitude and longitude values of the computational domain given in Figure 6.5 from the report, and the date of the data file, it was concluded that the job file delivered to GL was not the one used to produce the figures in the report.

Attempts to duplicate the report results continued. A number of runs were made with different job control parameter values that were inferred from the report. The original horizontal grid spacing was 10 km with 120 s for the long time step and 60 s for the small time step. In the technical report 5 km was used as the horizontal grid spacing. Since the number of the grid points (NNXP, NNYP) in the job file agrees with the contents of topographic data file, a comparison of the topographic data with an independent high resolution data set from Defense Mapping Agency was performed. The results indicate that with a 5 km resolution the two data sets compare favorably. This was taken as verification that the horizontal grid spacing was indeed 5 km. Once the grid spacing had been determined, the time steps were not short enough to satisfy the numerical stability criteria. Therefore, several tests were conducted to find the proper time length. When the time length was reduced to 10 s for both the long and small time scale, a lot of noise that had been evident in the results disappeared. Thus it was felt that 10 s was short enough to suppress the errors from the numerical integration. However, with this time scale a two hour simulation needs roughly 95 minutes CPU time to complete on a MicroVax III class computer system. It would take an even longer time to complete the simulation on a

MicroVax II class computer system, far in excess of the 1/5 real time reported by Cotton in the technical report.

The updated parameter values have now been set to those believed to correspond to the situation found in the technical report. Appendix B contains the most up to date version of the job file parameters, those which have been modified are highlighted with bold typeface. The x grid point location of the radar (IDR) for radial nudging should be 7 not 12. The local solar time at the beginning of simulation (STRTIM) is calculated using the average of the times for radar wind data and the sounding data. The specifications of the latitude of the southern edge (RLAT) and the longitude of western edge (WLON1) of the computational domain are read from Figure 6.5a in the Cotton report. The date of the simulation is different from the time of all three simulation cases; in Appendix A, the date of the first case is used. From the description in the report, the model should be run with the convective parameterization option activated. However, all attempts to use convective parameterization logic have resulted in a program execution failure. Queries of the Aster Corporation about this problem are still pending.

Model runs using the updated job file for two of the available CINDE cases were conducted. Similar graphical output were obtained but the results from the technical report cannot be reproduced. Horizontal cross-sections of the vertical velocity fields at 1 km above the surface for both cases are shown in Figures 2.2 and 2.3 respectively. The corresponding figures for the real data simulation of case 1 and case 2 in the Cotton report are Figures 6.2a-6.4a and Figures 6.6a-6.8a. The GL-3D model was run with topography, Cartesian grids and Cartesian nudging. The grid spacing is 5 km in horizontal. The depths of the vertical layers are 400m near the surface and stretch up to 750m near the top of the domain. The time interval is 10 s for both long and small time steps. The contour intervals are 0.01 cm/s in Figure 2.2 and 0.02 cm/s in Figure 2.3. The interval size is set automatically by the contour program depending on the range of data. The intervals plotted in the Cotton report are unknown and may vary from one figure to the next, this may contribute to the inability to duplicate the reported results.

2.5 Summary

A thorough analysis of the GL-3D model characteristics and run time environment has been performed. The separate modules of the GL-3D model have been assembled, compiled, linked and run on the AMPS. Initially the model was written in a non-standard programming language which required a preprocessor program to convert it to standard FORTRAN source code. Several tests were conducted to determine how the preprocessor

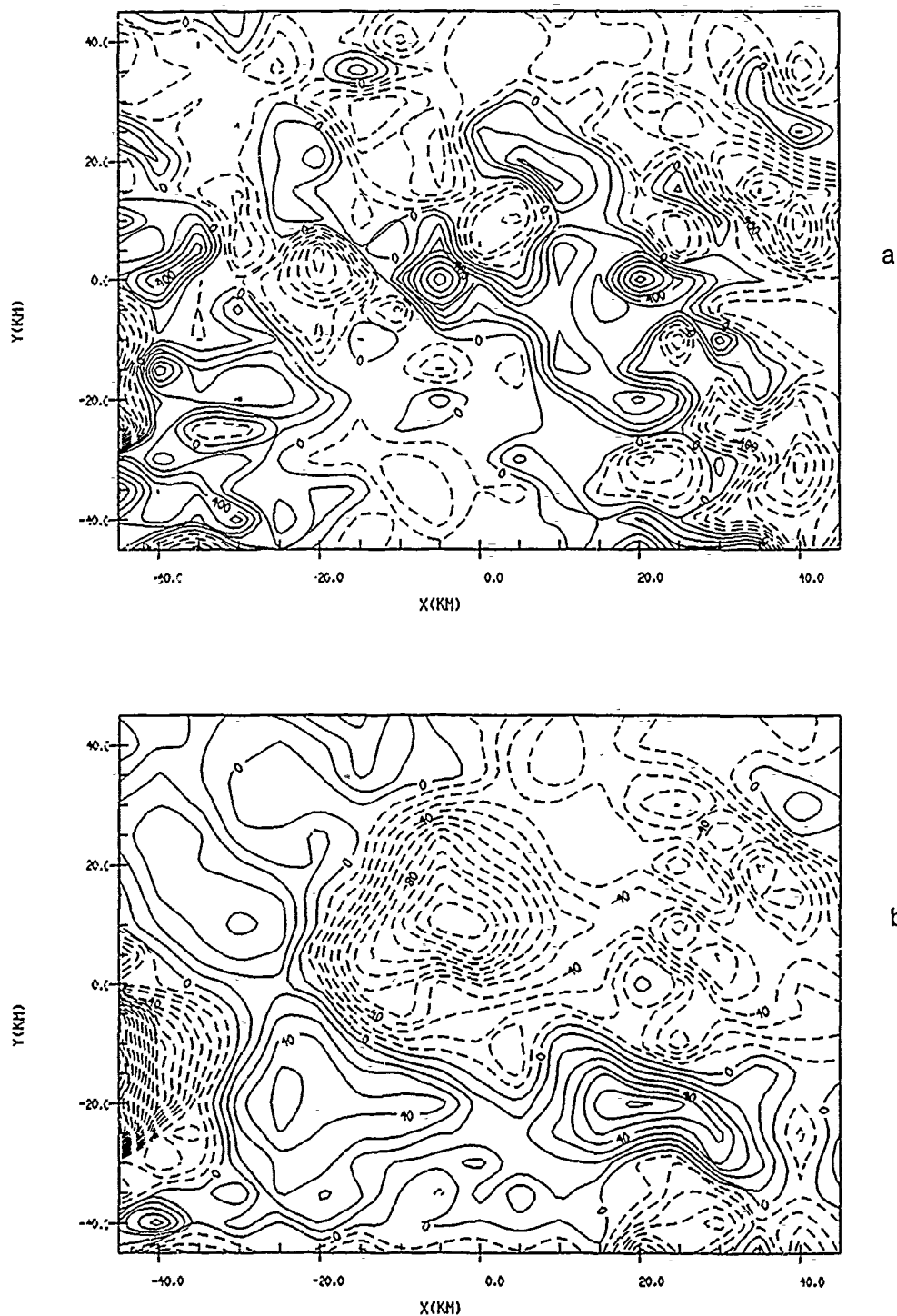


Figure 2.2 (a) Horizontal cross section of vertical velocity for case 1 at 1 km above the surface for model time = 1 hour; (b) same as (a) except model time = 1.5 hours.

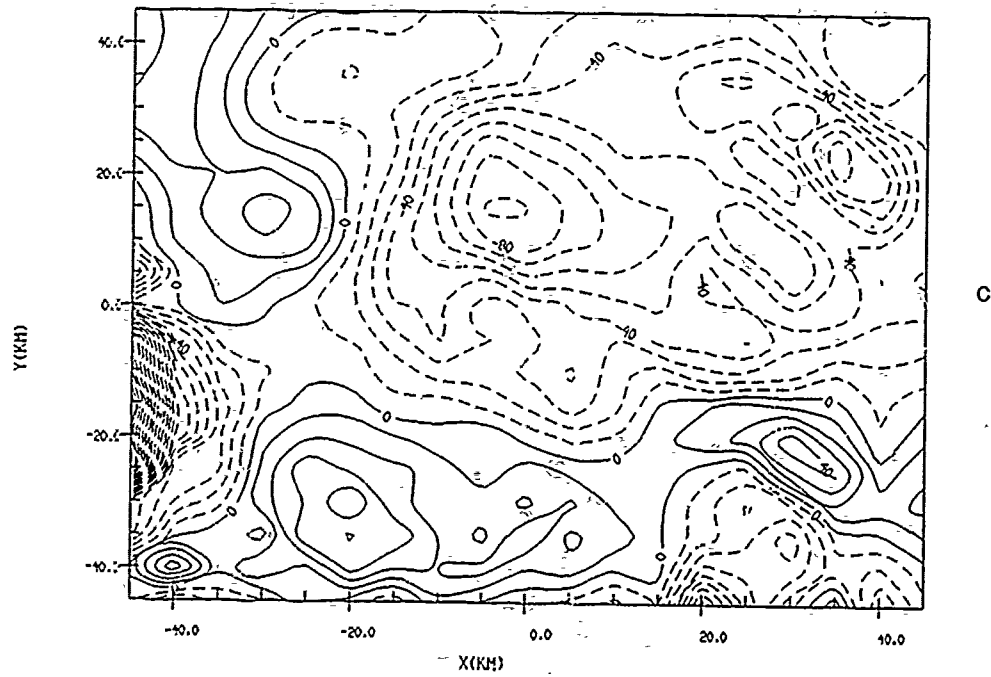


Figure 2.2 (continued) (c) same as (a) except model time = 2.0 hours.

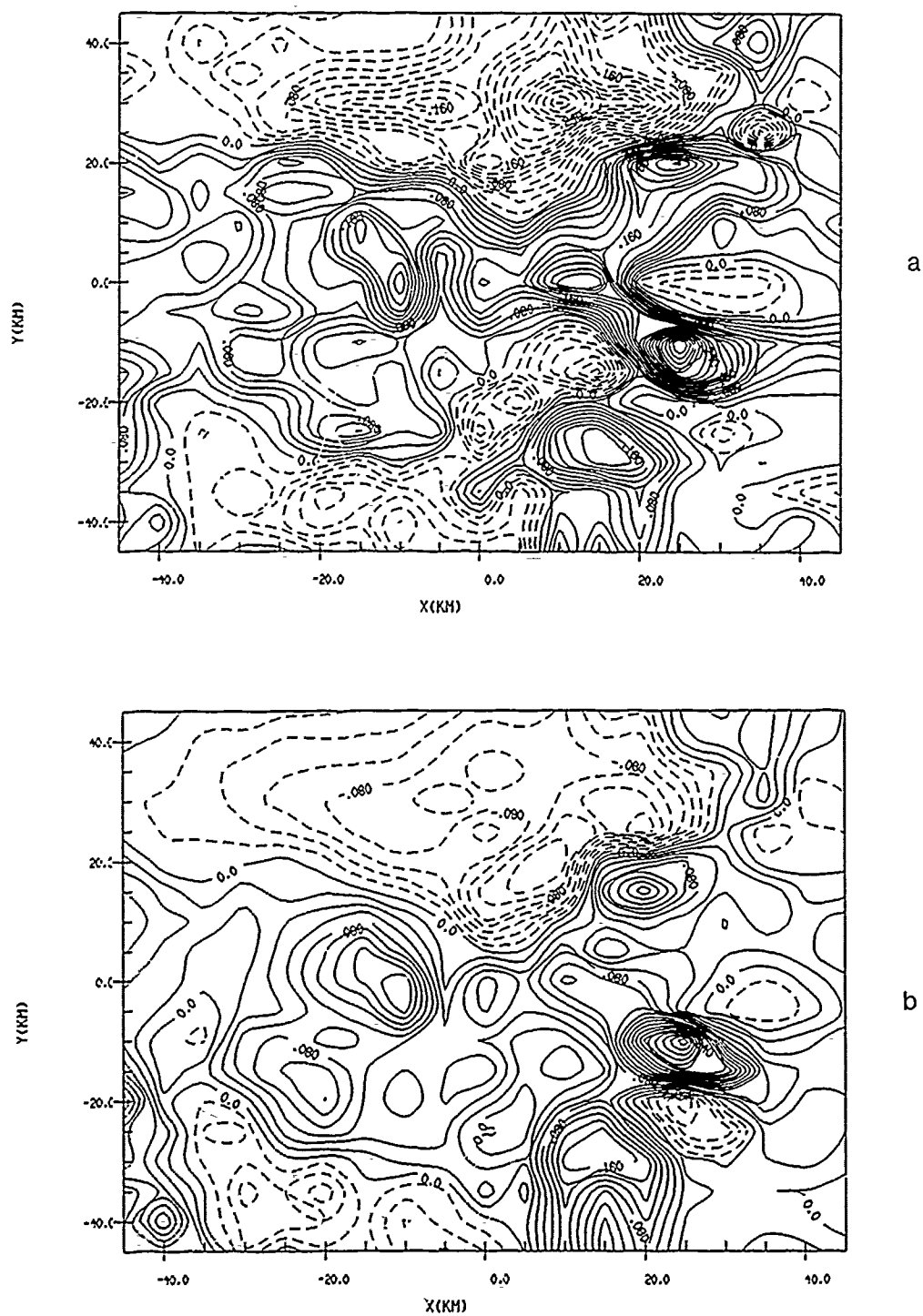


Figure 2.3 (a) Horizontal cross section of vertical velocity for case 2 at 1 km above the surface for model time = 1 hour; (b) same as (a) except model time = 1.5 hours.

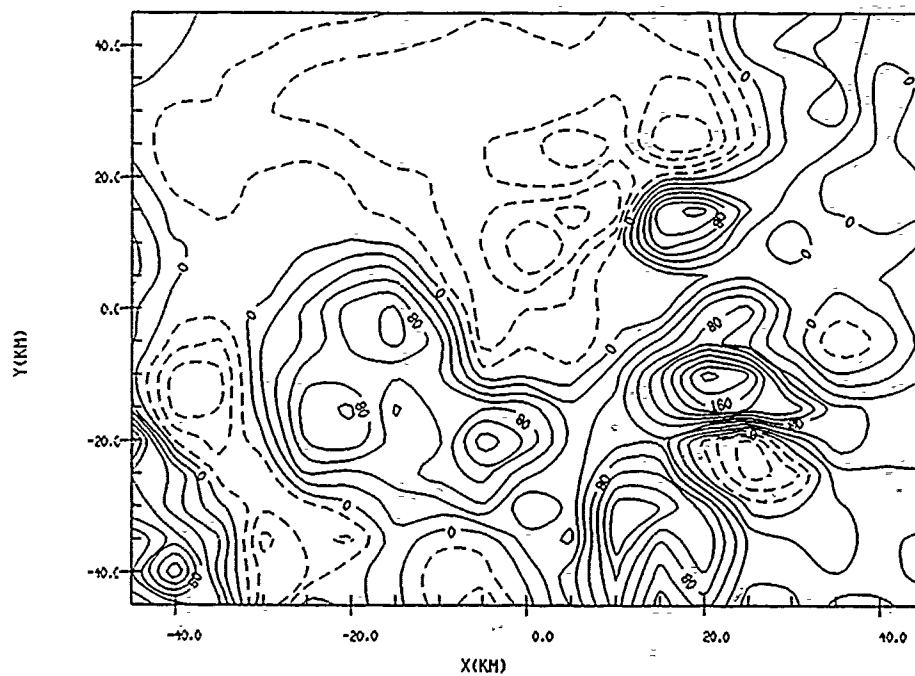


Figure 2.3 (continued) (c) same as (a) except model time = 2.0 hours.

worked, the knowledge obtained from those tests was used to develop a translator program that replaces the preprocessor. The translator was used to transform the GL-3D model source code into standard FORTRAN source code. A simple and easily understood preprocessor was written to interpret control characters embedded in the source code which implement user options. These control characters are retained by the translator because the options they control must each be run separately through a vectorizing compiler on machines with that capability. To prepare the model program for execution, three steps are followed:

- (1) run the MK_INCLUDES program to create the necessary include files with the selected run time options;
- (2) run the source code of GL-3D model through the AER preprocessor, compile and link to make an executable file;
- (3) run the GL-3D model with the specified input job control file.

A new output scheme for the GL-3D model has been developed which saves all three dimensional data at selected model times into an output data file. New display programs allow the user to interactively select and display horizontal or vertical cross sections of any field in the 3-D data file. The display programs aid interpretation of dynamic features of the model output through interactive display of multiple cross sectional analyses.

Several problems arose when test runs of the model were conducted. Some parameter values in the job control file needed to be modified to be made consistent with the test runs reported by Cotton et al. (1988). Problems encountered when attempting to execute the cumulus convective parameterization remain unresolved. While it appears that this part of the code was used to produce the results reported by Cotton, program execution errors have, so far, prevented its successful execution on AMPS. Many of the program modules have been analyzed and are understood, however, it is important to fully understand all aspects of the model before any modifications to the processing logic are introduced. Analysis of the GL-3D model source code and run time options should be continued in the future.

3. Nephanalysis Data Base

A data base management facility was developed on AIMS specifically for the RDNEPH program. Known as the Nephanalysis Data Base (NDB) it was designed to handle any field which has been mapped into the standard AFGWC RTNEPH grid system. NDB provides a comprehensive and consistent methodology for ingest, file management, access, modification, storage and archival of all nephanalysis data bases. The implementation on AIMS is provided through a library of FORTRAN callable subroutines and an overall utility program. The following functional requirements were identified for the RDNEPH data manager:

- data ingest into the data base;
- expansion of the data base to include new data types;
- support for both single (e.g., surface skin temperature) and multiple (e.g., upper air temperature - height - humidity) parameter data types;
- management of existing data files;
- both interactive and programmed access to the data base;
- modification of the data base; and
- archive and retrieval of data files

The data base design problem consisted of addressing each of these requirements in a logically consistent structure that exploited the strengths of the AIMS hardware and software environment. Toward this end a fundamental constraint is imposed on the data files, all fields must be formatted to AFGWC RTNEPH hemispheric grid. Adherence to the RTNEPH grid structure greatly reduces the scope of the data base management problem to one of handling regular gridded fields. This is contrasted, for example, to the problem of handling randomly scattered point data that was addressed in the separate AIMS Data Base (ADB) effort (Thomason and Ivaldi, 1990).

The largest obstacles to be overcome in the data base design were the size of the data files and the diversity of the data types. The RDNEPH program analyzes satellite imagery to produce an objective cloud analysis. Several supporting data bases are required to perform the analysis (see Section 4 for a discussion of the required data bases). Hemispheric projections of satellite data typically require 16 MB of mass storage per channel at the normal spatial resolution of 3 nm. RDNEPH program output often includes synthetic images at the same resolution as the sensor channel data. Thus it is common for data requirements to exceed several hundred megabytes for a single case.

3.1 NDB Data Base Structure

Data management capabilities are built around centralized data dictionaries which are maintained separately from the actual data files. A data dictionary is itself a specialized data base containing descriptive information such as file and data attributes that is used to characterize individual nephanalysis data files. Data dictionary records are designed to provide all information necessary to identify, select, and access a specific data file. Hence, all data management operations are performed on the centralized data dictionary independent of the data files themselves. Data files can be distributed anywhere on the system and are only accessed for actual data transfer operations. This greatly reduces the overhead of data management operations since the size of the data dictionary entries are small relative to the data files. It also reduces the complexity of the data base; since the large data files contain only gridded data with no additional descriptive information, the file structure can be formatted to exploit the natural structure of the underlying grid. RDNEPH investigations are based on case studies. To date, GL has on hand data for 9 different case study days. A logical separation between data from different case study days was built into NDB through creation of separate data dictionaries for each case day.

An interactive data base manager, such as NDB, has a fundamental requirement for speed. Interactive operations, by definition, must occur at least as fast as the user can assimilate the information. This was a prime concern during the design of NDB and the main driver behind the use of data dictionaries. Each record of the data dictionary describes one data file and is identified by a unique entry number. The data dictionary structure is optimized for fast access of individual file attribute information. Dictionary records are implemented as Indexed Sequential Access Method (ISAM) files keyed on data type and entry number. Record formats are predefined and include two sections; the first contains global information common to all data files and is fixed, the second contains type specific attributes and varies as a function of type. Table 3-1 provides a breakdown of the Data Dictionary structure. Global information includes the type of data stored, the location and resolution of the data, data access information, data file characteristics and format, and the archive location if any. The type specific section includes descriptors that pertain only to certain types of data, for example satellite sensor channel, satellite ID, spectral resolution, valid time, control flags, and pointers to data files which contain supplementary information (e.g., satellite ephemeris). The bulk of the data management function consists of maintaining, updating, and validating the data dictionary information.

Table 3-1 NDB data dictionary structure

Fixed

1)	Entry number	Unique entry number assigned to each data dictionary data.
2)	Data type	A numeric index that corresponds to a specific type of data. Type indices are stored in a separate table maintained by NDB.
3)	Grid Mesh	The data resolution in grid mesh coordinates defined by the AFGWC RTNEPH grid standard.
4)	Bytes/data element	The number of 8-bit bytes required to store one data element (i.e. one grid point).
5)	Version Number	Sequential numbers to support multiple versions of a single data file (e.g. skin temperatures derived from two different sources).
6)	Archive location	Tape and file number in NDB data archive.
7)	Grid Coverage	List of grid boxes that contain data.
8)	Data file name	Fully specified name of the file that contains the actual data.

Type Specific

Skin Temperature Valid Time		Pointer to supplementary file that contains the analysis valid time for each grid box in the surface temperature data file.
Satellite Imagery Channel		A numeric index that corresponds to a specific satellite sensor channel. Channel indices are stored in a separate table maintained by NDB.
	Spectral resolution	Number of bits needed to store the quantization levels of the satellite imagery data.
	Ephemeris	Pointer to supplementary file that contains ephemeris information for each grid box of the satellite imagery file.
Background Brightness Satellite ID		A numeric index that corresponds to a specific meteorological satellite. Satellite indices are stored in a separate table maintained by NDB.
	Channel	A numeric index that corresponds to a specific satellite sensor channel. Channel indices are stored in a separate table maintained by NDB.
Output Products Control Structure		A record number that points to the RDNEPH control structure used to generate the output file (see Section 6.1 for a discussion of control structures).
	Completion Flag	A control flag to indicate that the output file is complete (i.e. the RDNEPH run finished successfully).

All access to data in the nephanalysis data base is through NDB, no direct access of the files is supported. Data file structure is based on the inherent structure of the RTNEPH grid, each record contains one Neph box of data (except in cases where a single NEPH box exceeds the VMS record length limit of 32768 bytes). Standardized functions have been developed to access data on a Neph box level through either standard FORTRAN I/O operations or direct memory mapping. Memory mapping is possible since data files contain only data, no control information.

Data files can be maintained either on-line as an RMS file on any one of the active AIMS disks or as part of a tape archive. The on-line data dictionary tracks both types of files and queries can be performed regardless of where the file resides. The archive facility transfers files from disk to tape in a manner that maintains NDB control over the file. Data dictionary entries indicate where in the archive the file is stored (i.e., by tape and file number) and whether it is currently also available on disk. The NDB archive is stored on dedicated 8 mm high density tapes that can be accessed only through NDB routines.

3.1.1 AFGWC Polar Stereographic Grid System

The AFGWC RTNEPH grid is based on a hemispheric polar stereographic projection centered on either pole and true at a latitude of 60° . On the image plane formed by this projection lines of constant latitude are represented by concentric circles centered on the pole, meridians appear as straight lines radiating out from the pole with an angular spacing corresponding to the longitudinal spacing. The RTNEPH grid is formed by overlaying a regular Cartesian grid on the image plane with the X origin corresponding to the 10° East meridian and the Y origin to the 100° East meridian. A series of nested grids are formed by varying the grid spacing by factors of two from a whole mesh base resolution of 381 km. As resolution increases, the gridpoint resolution is referred to in terms of the fractional distance between grid points relative to whole mesh (e.g., half-mesh = 190.5 km, quarter-mesh = 95.25 km, etc.). Actual distance between grid points on the image surface varies with latitude ranging from 204 km at the equator to 408 km at the pole for a whole mesh grid. RDNEPH data fields are normally received already mapped into the AFGWC projection at a standard mesh resolution, however, AVHRR data obtained from NOAA NESDIS must be transformed from satellite scan projection. See Section 5 for a discussion of the image transformation procedure applied to AVHRR data.

3.2 NDB Run-Time Library

NDB functions were implemented in the RDNEPH applications development environment through creation of a run time library. Library functions have been developed to satisfy the functional requirements listed above and to perform all necessary data base operations. New data fields are entered into the data base, empty data files of the proper size and format created and initialized, and the data dictionary updated through a single NDB call. Identification and selection of a specific data file is performed through data dictionary query routines in the NDB library. Additional routines provide access to all or part of the selected data file contents through direct memory mapping or FORTRAN I/O operations.

NDB is accessible by any valid NEPHOIDS user. All library routines are identified by an "NDB_" prefix. The pathname of the NDB library is "NEF_NDB:NDB.OLB". Applications programs must initiate all data base operations with a call to the "NDB_OPEN" routine. Any program that calls NDB functions must include the VMS option file "NEF_SYSTEM:NEPH.OPT" in the link statement. An include file has been established to define the format of the data dictionary structure for use in applications programs. Both FORTRAN and C versions of the include file are available, they are NEF_INCLUDES:DD.INC and NEF_INCLUDES:DD.H respectively. Two levels of software documentation for the NDB utilities exist; an on-line users manual provides brief functional descriptions and calling argument syntax, in-code documentation provides detailed algorithm descriptions. The users manual is available in a file called NEF_NDB:NDB.DOC or through the VMS help facility (enter: \$NHELP LIBRARIES NDB). Source code for all routines is in the NEF_NDB directory.

Each NDB library routine returns a status message to indicate success or failure of the specific operation. Most routines are written in C, however, they can be called from any higher level programming language. Functional descriptions of individual NDB run-time routines are provided in Appendix D along with a sample program.

3.3 NDB Utility Program

A general purpose utility program called NDB is also available to NEPHOID users for interactive execution of common data base operations. The user interface follows the standard VMS DCL command syntax of a command verb followed by qualifiers and parameters through the use of the VMS Command Line Definition (CLD) facility. Prior to execution of the routine enter the command SET COMMAND NEF_CLD:NDB at the DCL

prompt to insert the program into the VMS Command Line Interpreter table, from then on simply type NDB with appropriate qualifiers to perform the desired operation. The NDB utility supports the following operations:

- selection of case study day;
- listing of available case study days;
- data base queries based on data type or entry number;
- brief or full data dictionary listings;
- data archive and restore;
- deletion of entries;
- imagery displays on Adage workstations;
- contour plots of selected data fields on any AIMS GKS graphics device; and
- generation of geographic maps for selected Neph boxes.

The NDB utility is described in the VMS on-line help facility (enter: \$NHELP NDB). NDB can be invoked from any terminal or workstation on AIMS and has access to all data files in the NDB data base. The query functions are useful for quickly examining the current contents of the data base and for determining whether a particular file exists and whether it is stored on-line or in the tape archive. Display functions can also be used for quick quality control checks on the data through image or contour plot displays. Appendix E contains an NDB utility user's guide.

4. RDNEPH Data Ingest Utilities

Most data used in the RDNEPH project are obtained through periodic data saves that are performed on the operational global RTNEPH data bases at AFGWC. RTNEPH data bases are updated constantly whenever new data are obtained. The save process requires that a static copy of the data be dumped to a backup computer and then transferred to tape. Saves are made for each of the data bases that are input by the RTNEPH and for the nephanalysis itself. Every effort is made to insure that all data bases are synchronized on the valid time of the nephanalysis run, however, this is not always possible since it is imperative that the save does not interfere with the operational programs at AFGWC. The data tapes are copies of the AFGWC data base format packed in Unisys save format. Since there are significant differences between the AFGWC and NDB formats and the Unisys and AIMS word size and data encoding schemes, it is necessary to use conversion programs to translate the data. Separate conversion programs have been developed for each of the input data bases. They read the data from tape, decode the Unisys run length encoding, reformat the data to NDB specifications, and add the data to the case study data base.

There are two primary considerations that must be accounted for when decoding packed UNISYS data on a VAX system: 1) UNISYS uses a 6-bit byte and a 6-byte 36-bit word, VAX uses an 8-bit byte and a 4-byte 32-bit word; and 2) UNISYS packs words from left to right (i.e. least significant bit to most significant bit) while VAX is right to left. The former problem is accounted for through the use of a C data structure wherein 16 UNISYS bytes are equivalenced to 12 VAX bytes. The reason that a C structure is used is that C supports data fields that can be defined as any number of bits in length (up to 32). By defining 6 bit data fields in the VAX data structure, the process of remapping the packed UNISYS data is greatly simplified. Table 4-1 illustrates how 16 UNISYS bytes are mapped to the VAX data structure. The problem of reverse bit order is compensated for through the use of an AIMS library routine called SWPBYTE that reorders bytes from 4-3-2-1 to 1-2-3-4 on the VAX.

4.1 Satellite Global Data Base

Satellite sensor data are received from AFGWC in a format known as the Satellite Global Data Base (SGDB). An SGDB record contains one whole mesh grid box of satellite data at 64th mesh resolution for one of the OLS channels, alternate records contain alternate channels (e.g. record x contains IR data for whole mesh box n, record x+1

Table 4-1 Mapping of 6-bit UNISYS data to VAX data structure

<u>VAX Field</u>	<u>size (bits)</u>	<u>UNISYS byte number (1-16)</u>
1	2	left 2 bits of 6
2	6	5
3	6	4
4	6	3
5	6	2
6	6	1
7	4	left 4 bits of 11
8	6	10
9	6	9
10	6	8
11	6	7
12	4	right 4 bits of 6
13	6	16
14	6	15
15	6	14
16	6	13
17	6	12
18	6	left 2 bits of 11

contains visual data for box n, record x+2 has IR data for box n+1, etc.). In addition to sensor data, ephemeris information is also included in the data record at 4th mesh resolution (i.e., 1 ephemeris entry for each 16X16 array of sensor data). Sensor data are packed as 6-bit values by 64th mesh row across a whole mesh box (i.e., 64 rows with 64 data points per row). To accommodate the 36-bit UNISYS word size, 11 words (36-bit) are used to store each row such that the first 64 bytes (6-bit) contain data and the last 2 are empty.

After decoding, sensor data are stored on AIMS as 8-bit values (6-bit AFGWC data are shifted 2 bits left) with low numbers representing warm/dark conditions. For each SGDB received from AFGWC, four entries are made in the Neph data base, one each for the two satellite sensor channels at 64th mesh, and two ephemeris files (one for each sensor channel) at 4th mesh. Complementary sensor and ephemeris files are related by the EPHEM_ID field in the sensor data dictionary entry in NDB. Data are stored by Neph box in column major format, sensor data require 1 byte (8-bit) per grid point, ephemeris data are stored in 16 byte data structures following the format in Table 4-2.

Table 4-2 Satellite ephemeris data structure.

<u>Field</u>	<u>Size (bytes)</u>	<u>Description</u>
ID	1	Satellite ID
QO	1	Quarter orbit number (1-4)
IREV	2	Revolution number (+) NH, (-) SH
YYDDD	4	Year and Julian day
TIME	2	Universal time (HHMM)
LOOK	1	Look angle (deg) (+) right, (-) left
ZENITH	2	Zenith angle (deg)
AZIMUTH	2	Azimuth angle (deg)
FLAG	1	Timely data flag

4.2 Upper Air Temperature and Humidity

AFGWC upper air data formats have changed over the years, the current format includes temperature and height information for 10 mandatory pressure levels (1000 - 100 mb), specific humidity for the surface and 6 mandatory levels (1000 - 300 mb), and tropopause height. Upper air data are at whole mesh resolution and are packed on tape one 36-bit word for each level/parameter combination for a total of 28 words per whole mesh box (Table 4-3). Temperatures are stored as degrees K*10, Heights as meters*10, and humidity as grams/kilogram*100.

Table 4-3 AFGWC upper air data packing format by 36-bit word number.

<u>Level(mb)</u>	<u>Temperature</u>	<u>Height</u>	<u>Humidity</u>
Sfc			21
1000	1	11	22
850	2	12	23
700	3	13	24
500	4	14	25
400	5	15	26
300	6	16	27
250	7	17	
200	8	18	
150	9	19	
100	10	20	
Trop		28	

The RDNEPH data format combines temperature, height, and humidity data into one whole mesh structure for 11 pressure levels and the surface. Data are stored such that all information for a given level are adjacent in memory (Table 4-4). This simplifies data

access by the applications software where typically all information for one level is used together.

Table 4-4 RDNEPH upper air data structure

<u>Field</u>	<u>Size (bytes)</u>	<u>Description</u>
UA_TEMP	2	Temperature (K*10)
UA_HGT	2	Height (m*10)
UA_SHUM	2	Specific Humidity (g/kg*100)

4.3 Surface Temperature

Hemispheric surface temperature analyses are received at 8th mesh resolution in sets of three. The three data sets represent two analyses and one forecast separated in time by three hours each. The analyses are derived from separate reports of sea surface temperature obtained from the Navy and of surface (shelter) temperatures from conventional surface observations. For each Neph grid box of data, the 8th mesh analyses are tagged with the valid times of the sea surface observations, the land surface observations, and the base time of the analysis. AFGWC tapes are packed three 12-bit values per 36-bit word with each 12-bit value corresponding to either an analysis or forecast temperature in degrees K*10. The first 4096 words of each tape record contain the three 8th mesh temperature fields for one Neph box, the following three words contain the valid times for the sea surface temperatures, the land surface temperatures and the analysis times respectively. Time information is packed as an 18-bit value (2 per word) with the high order bits corresponding to the first analysis, and the low order bits corresponding to the second analysis. The forecast valid time is found by adding three hours to the most recent analysis base time. Table 4-5 summarizes the surface temperature tape packing format.

The RDNEPH surface temperature format consists of a separate entry in the NDB data base for each temperature field. Thus from a single tape three NDB entries will be created. Each 8th mesh temperature value is stored as a 2-byte VAX integer (I*2) as degrees K*10. Three supporting data files containing the analysis base times for each Neph grid box are also created from the AFGWC tape. Base time files are at Neph grid resolution and are related to the corresponding 8th mesh temperature files by the TIME_ID field in the temperature data dictionary entry in NDB. RDNEPH time interpolation is performed at Neph grid resolution using the base time information in these supporting data

Table 4-5 AFGWC surface temperature data packing format by 36-bit word.
Note: bit 0 is MSB, 35 is LSB

Word	Bit range	Description
1-4096	0-11 12-23 24-35	Temperature analysis 1 (K*10) Temperature analysis 2 (K*10) 3 hour temperature forecast (K*10)
4097		Label name: 00STnn for Northern hemisphere SHSTnn for Southern hemisphere where nn = Neph box number
4098		not used
4099	0-17 18-35	Sea temperature valid time used in analysis 1 Sea temperature valid time used in analysis 2
4100	0-17 18-35	Land temperature valid time used in analysis 1 Land temperature valid time used in analysis 2
4101	0-17 18-35	Base time of analysis 1 Base time of analysis 2

files. For case study days when more than one surface temperature tape is provided, unpacking occurs such that the most recent data files overwrite any older files with the same valid time, and analyses always supersede forecasts.

4.4 Background Brightness

Background brightness values are provided from the AFGWC data base at 8th mesh resolution for up to four satellites. This data base is intended to provide a clear column digital count for the OLS visible sensor for a specified location on the Earth. It is used by the visible processor to determine the cloud/no-cloud threshold and by the infrared processor to modify the cloud/no-cloud threshold for highly variable backgrounds. To insure that the information reflects seasonal changes in background and solar illumination, the background brightness data base is updated whenever a clear scene is detected by the RTNEPH. Updates are performed through a weighted average of the existing data base value and the observed OLS visible count.

Like the upper air data, the format of the AFGWC background brightness data base has changed with time. The present format consists of data for each 8th mesh grid point for four satellites. Satellites are identified in the STOKER data base in terms of their relative position in the background brightness file. For each Neph grid box, the data base

consists of 8192 words containing the background brightness values and 5 words of fiducial information. Snow information is included in the brightness data, brightness values default to a fixed value (also defined in STOKER) for snow covered land and ice covered water. Ice covered grid points are determined by examining the geography data base described in Section 4.5. Table 4-6 describes the AFGWC tape format for the background brightness data base.

Table 4-6 AFGWC background brightness data packing format by 36-bit word.

<u>Word</u>	<u>Bit range</u>	<u>Description</u>
1-4096	0-3	Snow Flag from SNODEP
	18-24	Background brightness for satellite 2
	25	Update flag for satellite 2
	27-33	Background brightness for satellite 1
	34	Update flag for satellite 1
4097-8192	18-24	Background brightness for satellite 4
	25	Update flag for satellite 4
	27-33	Background brightness for satellite 3
	34	Update flag for satellite 3
8193		Label name: 00SCnn for Northern hemisphere SHSCnn for Southern hemisphere where nn = Neph box number
8194-8196		not used
8197		Valid time of most current data (DDDDHH)

The RDNEPH background brightness data base consists of a separate entry in the NDB data dictionary for each satellite. The file format is a one byte (8-bit) entry for each 8th mesh point. The range of data values currently reflects the 6-bit resolution of the SGDB data (i.e. 0-63), however it can be expanded to 8-bit (i.e. 0-255) without modification. Above 8-bit it will be necessary to increase the word size for each grid point from 8-bits to 16-bits. The satellite ID and channel number associated with each background brightness file is contained in the NDB data dictionary entry as the ISAT and BCHAN fields respectively.

4.5 Terrain Heights and Geography Type

The AFGWC 8th mesh terrain and geography fields are used by the RDNEPH to assign low cloud altitudes and to determine the magnitude of the cloud/no-cloud threshold. Four geography types are identified in the geography field: 1) water, 2) land, 3) ice over land, and 4) coastline. Terrain heights are maintained in both 10 m and 100 ft increments.

Tape packing requires a single 36-bit Unisys word for each 8th mesh point according to the format in Table 4-7.

Table 4-7 AFGWC terrain and geography data packing format by 36-bit word.

<u>Word</u>	<u>Bit range</u>	<u>Description</u>
1-4096	0-9	Terrain height in 100 ft increments
	10-20	Terrain height in 10 m increments
	21-29	not used
	30-35	Geography type: 1 - water
		2 - land
3 - ice over land		
4097	4 - coastline	
	63 - off hemisphere	
	Label name: PGA8nn for Northern hemisphere	
4098	SH8Tnn for Southern hemisphere	
	where nn = Neph box number	
4099	Valid time of most current data (DDDHH)	
4101	not used	

The RDNEPH data base maintains two separate entries for terrain and geography. Geography type is carried in an 8-bit byte for each grid point, with the same numerical mapping as in Table 4-7 (i.e., 1-water, 2-land, etc.). Terrain heights are stored as 16-bit words containing heights in 10 m increments.

5. AVHRR Mapping Utility

The Advanced Very High Resolution Radiometer (AVHRR) Mapping Utility was developed as part of the Research and Development Nephanalysis (RDNEPH) project at the Geophysics Laboratory (GL). Its purpose is to provide a means of transforming AVHRR satellite data from raw scan line format to the Air Force Global Weather Center (AFGWC) Satellite Global Data Base (SGDB) standard projection. This standard is used for all data in the RDNEPH project. The adoption of the SGDB data format standard eliminates many problems inherent in the use of raw scan line data format. These problems include: 1) comparing AVHRR data with imagery data from other satellite sensors, such as the Operational Line Scan (OLS), because of the different formats used for each sensor, 2) combining the satellite data with data from other sources such as surface temperature data, again because of different formats and 3) the fact that only a single orbit of data can be displayed at one time. In the SGDB, all satellite data are reformatted from scan line format to the AFGWC Real Time Nephanalysis (RTNEPH) grid system. This grid system is based on a hemispheric polar stereographic map projection centered at the pole and true at a latitude of 60 degrees. By reformatting data to this grid, a polar stereographic projection of the data is created. Multiple satellite orbits are mosaiced to one grid, allowing contiguous orbits to be displayed as one image. Currently available at GL are nine tapes of AVHRR data containing coverage of the entire Northern hemisphere. The goal of the remapping effort was to develop software to transform the data from the tape scan line format to the SGDB format for use in the RDNEPH project.

An algorithm has been developed which transforms satellite data from scan line format to the RTNEPH grid. In this report, the term mapping will refer to the steps necessary to transform the data from scan line format to the RTNEPH grid. In theory, the mapping technique is straightforward. The mapping process for AVHRR sensor and ephemeris data requires seven Northern Hemispheric RTNEPH grids. An RTNEPH grid is a Cartesian coordinate grid superimposed on a polar stereographic map projection of the Northern hemisphere such that each (x,y) point in the grid is associated with a specific latitude/longitude (lat/lon) point in the map projection (see Figure 5.1). A grid can be visualized as a 2-dimensional array where each array element contains information valid at a specific lat/lon location. A complete discussion of the RTNEPH Grid system is given in Sections 3.1.1 and 5.3. The first of the seven grids which must be created is a location grid; it contains the actual lat/lon coordinates of each grid point. The lat/lon coordinates are calculated from the (x,y) coordinates using the transformation given in Section 5.2.1. The other six grids, which have a one-to-one correspondence with the location grid, store the

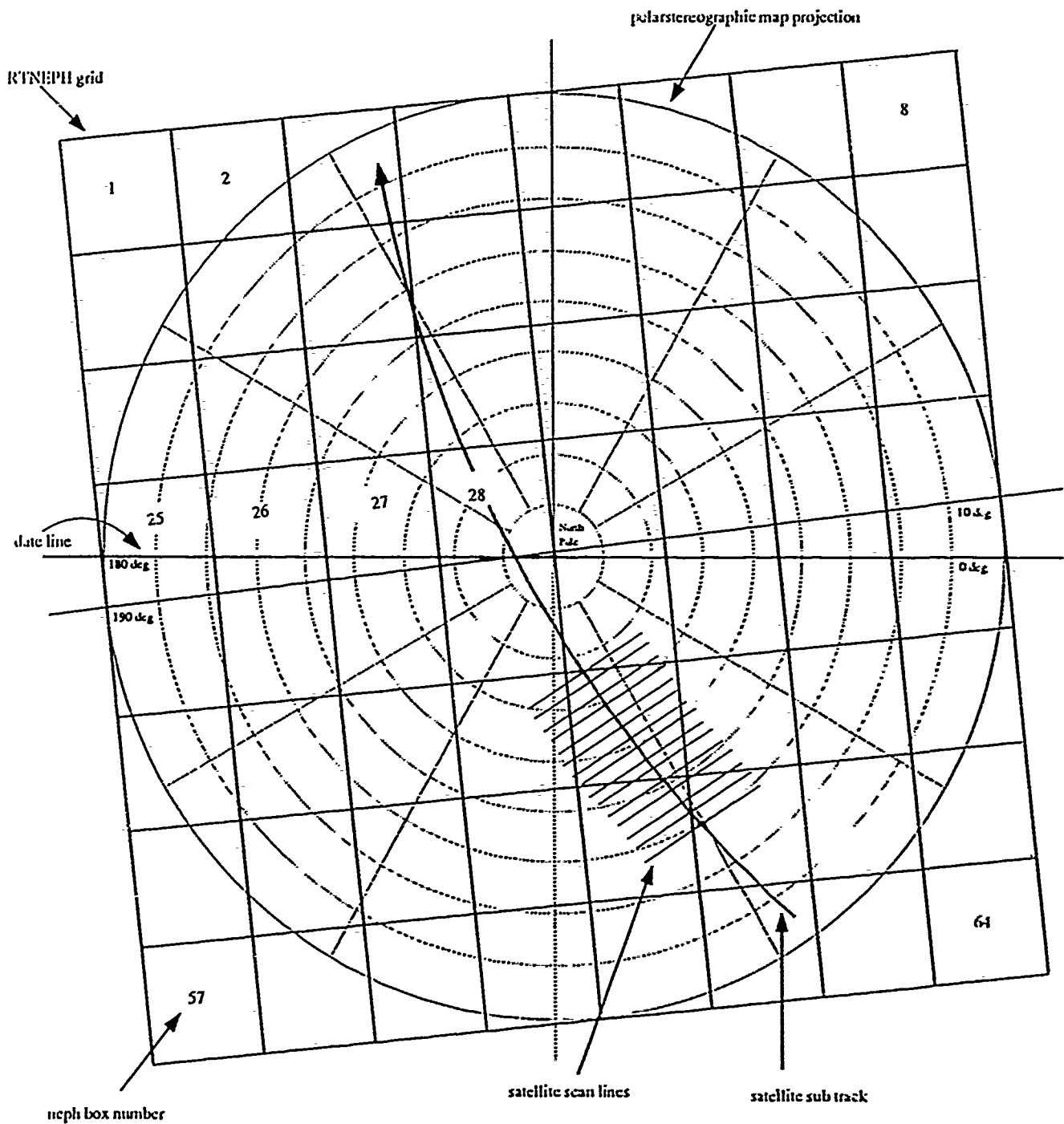


Figure 5.1 The AFGWC RTNEPH grid system. The grid is superimposed on a polarstereographic map projection of the Northern Hemisphere.

mapped data from the five AVHRR sensor channels and the associated ephemeris data. Throughout this report, the term grid will specifically mean the location grid; grid point will mean a location grid point in lat/lon coordinates. The grids containing the mapped sensor and ephemeris information will be called the sensor grids and ephemeris grid respectively.

In the next step of the mapping process, the AVHRR satellite data in raw scan format that are to be mapped are selected. Each data set includes earth location, sensor and ephemeris information. The earth location information gives a latitude/longitude coordinate for each satellite data point. A description of the AVHRR data is found in Section 5.1. Keying on the separate latitude/longitude information associated with both the grid and the satellite data, the actual mapping process works as follows. The location grid is processed, point by point. First, the geographical range of the satellite lat/lon data is calculated to determine if it encompasses the grid point location. If it does, a search of the satellite lat/lon data is done to find the data point which is spatially closest, or the nearest neighbor, to the grid point. A criterion based on minimum Euclidean distance is used as the measure of spatial closeness. When a nearest neighbor is found, the sensor and ephemeris data associated with this satellite lat/lon location are mapped to the sensor and ephemeris grid locations corresponding to the grid point. If the satellite data are not geographically close to the grid point location, the search is not done and no data are mapped for this grid point. When the entire location grid has been processed, the mapping is complete. If data are available such that a data point is mapped to every grid point, a complete hemispheric map will result.

Key to the mapping algorithm are two search methods used to identify the nearest neighbor data points. The first method is a binary search technique. This technique is preferred due to its speed and efficiency. However, it is not always applicable as it relies on certain assumptions about the satellite latitude/longitude data. When these assumptions fail, an alternative search method based on a random search is used. The binary search technique and required assumptions are detailed in Section 5.4.1. The random search method is discussed in Section 5.4.2.

In practice, the mapping process becomes complicated. A major issue is the large amount of data that must be mapped. The satellite data set used in this project consists of approximately 40 Mbytes of data. The RTNEPH hemispheric grid used is a sixty-fourth mesh grid which contains (4096×4096) or 16,777,216 points. A search of the satellite data must be done for each of the grid points. This size problem and how it is handled in the mapping utility is discussed in Section 5.5. Certain attributes of the data and of the grid system further complicate the mapping process. Basic assumptions about the data,

essential to the binary search algorithm, fail to hold uniformly. Also, the RTNEPH grid design and the convention of defining longitudes on a -180 to +180 degree base create additional problems. These data related issues and their solutions are discussed in Section 5.6.

5.1 AVHRR Data

The mapping utility is designed specifically to map AVHRR Global Area Coverage (GAC) satellite data which has a nominal resolution of 4 km. This project used a data set consisting of nine quarter orbits over the Northern hemisphere from date 82162. These data provide enough satellite coverage to create a complete polar stereographic map of the Northern hemisphere; in the data, latitudes range from 0 to 90 degrees North; longitudes range from 0 to -180 degrees West and 0 to +180 degrees East. Consecutive orbits have duplicate scan coverage, as the satellite approaches the higher latitudes the overlap increases (see Figure 5.2). For consistency, the final grid was constructed from the most recent information. The data were processed in order of time collected and newer data values overwrote previously mapped values. Note that time checks are not done in the utility. The user must order the data by time before processing.

Each orbit contains on the order of 7000 scan lines; each scan line is comprised of 409 scan elements. Associated with each scan element are imagery data from the five AVHRR sensor channels, an earth location in latitude/longitude coordinates and some ephemeris information. These data are obtained from NOAA NESDIS stored on tape in packed binary format. The data must be unpacked and written to disk prior to processing. This is accomplished using an existing AVHRR data unpacking utility which writes the data to the following files:

Ch1_alb.dat } Ch2_alb.dat }	Normalized Albedoes (reflectivities) for channels 1 and 2
Ch3_temp.dat } Ch4_temp.dat } Ch5_temp.dat }	Brightness Temperatures for channels 3, 4, and 5
All_loc.dat	Earth Locations (latitude,longitude)
All_solzen.dat	Solar Zenith Angles
All_time.dat	Scan Line Times

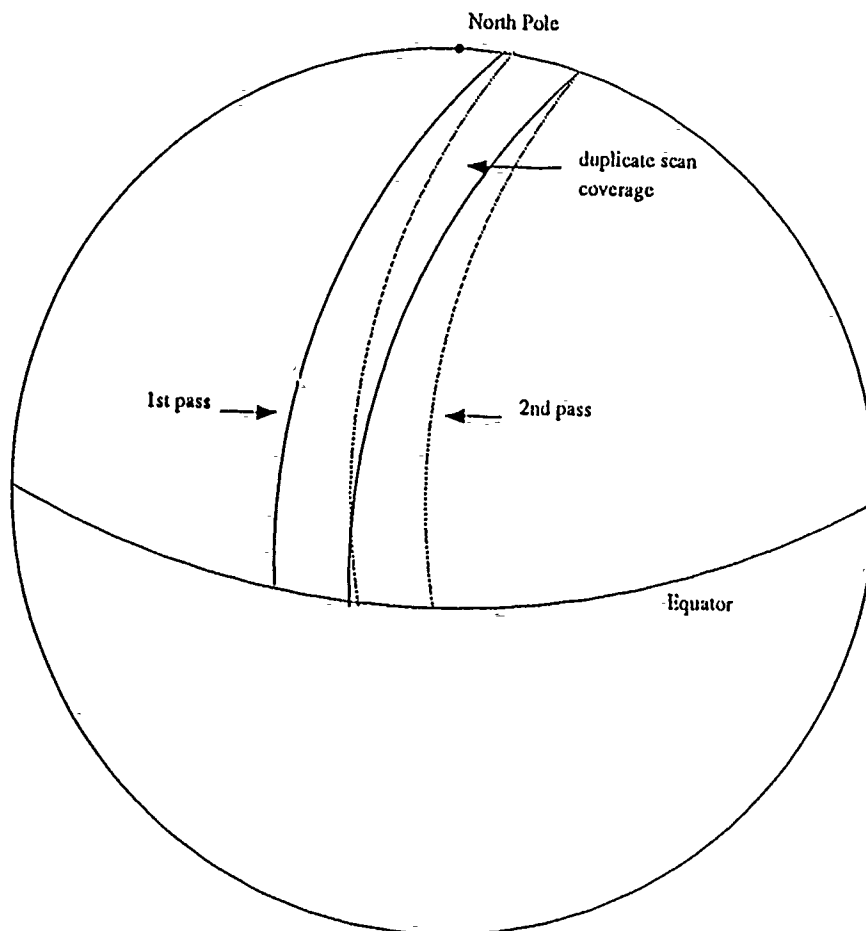


Figure 5.2 Schematic of consecutive satellite orbits depicting duplicate scan coverage. This figure is not to scale.

The unpacking utility files are constructed in scan line format. The files are essentially large data arrays where each array row corresponds directly to a scan line and each row element is a data point. The term scan column refers to the data points which correspond directly to a column in the data array. This format is maintained in the mapping utility. An important result of this format is the one-to-one correspondence which exists between the elements of the individual sensor, ephemeris, and location arrays. For example, the solar zenith angle of the second data point in the first scan line is found at the array address (1,2) of All_solzen.dat. The Channel 1 albedo of this same point is found at (1,2) of Ch1_alb.dat, etc. This array correspondence reduces the work needed to map the data because the search for the nearest neighbor points is limited to the earth location file, All_loc.dat. All_loc.dat is searched until a nearest neighbor point is identified and the array address of this point is stored. The ephemeris and sensor information associated with this point are found at the same array address in the files All_solzen.dat, Ch1_alb.dat, etc.

The sensor information collected by the satellite is stored in terms of albedoes and brightness temperatures. For imaging and display purposes, the data are converted to 8-bit grayshade scale with digital values ranging from 0 to 255. The following conversion conventions are used:

Channels 1 and 2 (Reflected solar) - stored albedoes range from 0 to 100.

$$gs = albedo * 2.55$$

Channels 3,4 and 5 (Infrared) - stored brightness temperatures range from 190 to 320 degrees K. Any values outside this range are treated as either 190 degrees or 320 degrees.

$$gs = 255 - (br\ temp - 190) * 255/130.$$

5.2 RTNEPH Grid

In accordance with the SGDB standard adopted for all data in this project, the RTNEPH grid used is a 64th mesh grid. At this mesh, the base resolution of the grid is approximately 6 km. This resolution is comparable to the nominal AVHRR sensor resolution of 4 km. However, there is not a one to one correspondence between the grid points and the satellite data points (see Figure 5.3). This raises the issue of how to map the

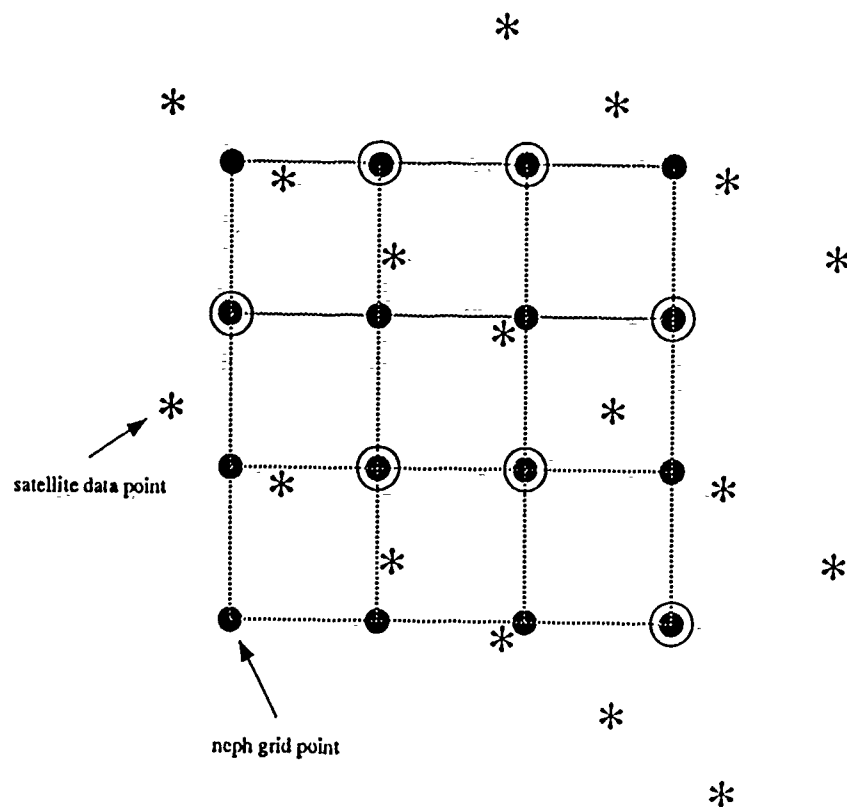


Figure 5.3 Correspondence between a Neph grid and satellite data. No data would be mapped to the circled grid points if each satellite data point was mapped to the nearest grid point. Instead, the nearest satellite data point is mapped to each grid point.

satellite data to the grid. One approach is to process the satellite location data, using the lat/lon information to map each data point to the nearest grid point. However, this method does not ensure that each grid point will receive data as shown in Figure 5.3. Instead, the inverse approach is used. The location grid is processed point by point. The satellite lat/lon data are searched to find the data point which is closest to each grid point. This nearest data point is mapped to the grid. With this approach, a single satellite data point may be mapped to two grid points or may not be mapped to the grid at all. But, a completely filled grid is guaranteed.

5.2.1 Coordinate Transformation Equations

In the RTNEPH grid point array, each element is designated by an (i,j) coordinate pair. The (i,j) indexing convention is left-handed for the polar stereographic grid: the column index i increases from 1 on the left side toward the right, whereas the row index j increases from 1 at the top of the grid toward the bottom. The grid point Cartesian coordinates are expressed relative to the pole on which the projection is centered according to:

$$x = i - ip, \quad (5.3.1)$$

$$y = -H(j - jp). \quad (5.3.2)$$

where ip and jp are the i and j coordinates of the pole and H designates the hemisphere of the projection. (Northern Hemisphere: $H = +1$, Southern Hemisphere: $H = -1$). At 64th mesh the polar location (ip,jp) is at (2049,2049).

Since the grid is superimposed on a Northern hemispheric map projection, a corresponding latitude and longitude can be calculated directly for each (i,j) pair resulting in a complete hemispheric grid in a lat/lon coordinate system. This location grid is constructed using the following coordinate transformation:

The latitude of any grid point is given by

$$\phi = H \arcsin \left\{ \frac{\left(\frac{a}{d_o SM} \right)^2 (1 + H \sin \phi_o)^2 - (X^2 + Y^2)}{\left(\frac{a}{d_o SM} \right)^2 (1 + H \sin \phi_o)^2 + (X^2 + Y^2)} \right\} \quad (5.3.3)$$

- where ϕ_o = 60 degrees N the standard latitude at which the polar stereographic projection is true
 a = 6371.2213 km Earth radius based on a sphere having the same volume as the Earth
 d_o = 381 km distance between whole-mesh grid points at X,Y - grid point Cartesian coordinates defined in (5.3.1) and (5.3.2)
 M = mesh factor, at 64th mesh, $M = 1/64$

The longitude is given by

$$\lambda = \begin{cases} \lambda_o + \arccos \left(\frac{x}{(x^2 + y^2)^{1/2}} \right), & Y \geq 0 \\ \lambda_o - \arccos \left(\frac{x}{(x^2 + y^2)^{1/2}} \right), & Y \leq 0 \\ \lambda_o \text{ (by definition)}, & X = Y = 0 \end{cases} \quad (5.3.4)$$

where: $\lambda_o = 10$ degrees East.

5.3 Calculation of Ephemeris Data

In addition to the sensor data at 64th mesh resolution, ephemeris information are maintained at quarter mesh resolution. The structure of the ephemeris data is given in Table 4-2. All data fields except the look and azimuth angles are available directly from the

tape information. Look angle is easily computed from the relative element number along a scan line. Azimuth angle (α) is computed from time and location information as follows:

for $\psi > 0$

$$\cos(\alpha) = (\sin(\phi_s) - \sin(\phi) * \cos(\psi)) / (\cos(\phi) * \sin(\psi))$$

$$\sin(\alpha) = -\cos(\phi_s) * \sin(\lambda_s - \lambda) / \sin(\psi)$$

for $\psi < 0$

$$\cos(\alpha') = (\sin(\phi_s) - \sin(\phi') * \cos(\psi)) / (-\cos(\psi') * \sin(\psi))$$

$$\sin(\alpha') = (\cos(\phi_s) * \sin(\lambda_s - \lambda') / \sin(\psi))$$

where: ψ is the geocentric angle between satellite subpoint and viewed point, ϕ the geographic latitude of viewed point (scan element), λ the geographic longitude of viewed point, ϕ_s the geographic latitude of satellite subtrack, and λ_s the geographic longitude of satellite subtrack. The quadrant of α and α' is defined by the sine and cosine relationships for each case (see Figure 5.4). The calculation of α is very sensitive to changes in ψ . Since ψ itself is calculated, accuracy errors may result leading to severe errors in the azimuth angle. To avoid this ψ is adjusted according to strict criteria until the value of α becomes stable.

5.4 Search Methods, Nearest Neighbor Criterion

Basic to the mapping utility are the binary search and the random search routines which locate the AVHRR latitude, longitude data point which is spatially closest to each grid point. A search of the AVHRR location array must be done for each grid point in the location grid. Two different search techniques are used as neither technique performs satisfactorily in all situations. The simpler search method is a random search in which the Euclidean distance between the grid point and every data point is calculated and the data point with minimum distance is chosen (see Section 5.4.2). This method works in most cases; its failures are detailed in Section 5.6. However, it is not practical computationally given the large amount of data and the grid size. A faster, more efficient binary search method has been devised which greatly reduces the computational burden and the number of distance comparisons which must be made. It utilizes the fact that as a result of the satellite scanning geometry, latitude and longitude change monotonically across the scan lines and down the scan columns (See Figure 5.5). This assumption is the key to the speed and efficiency of the binary technique. Recall from Section 5.1 that the array rows and columns of All_loc.dat correspond directly with the scan lines and scan columns. This

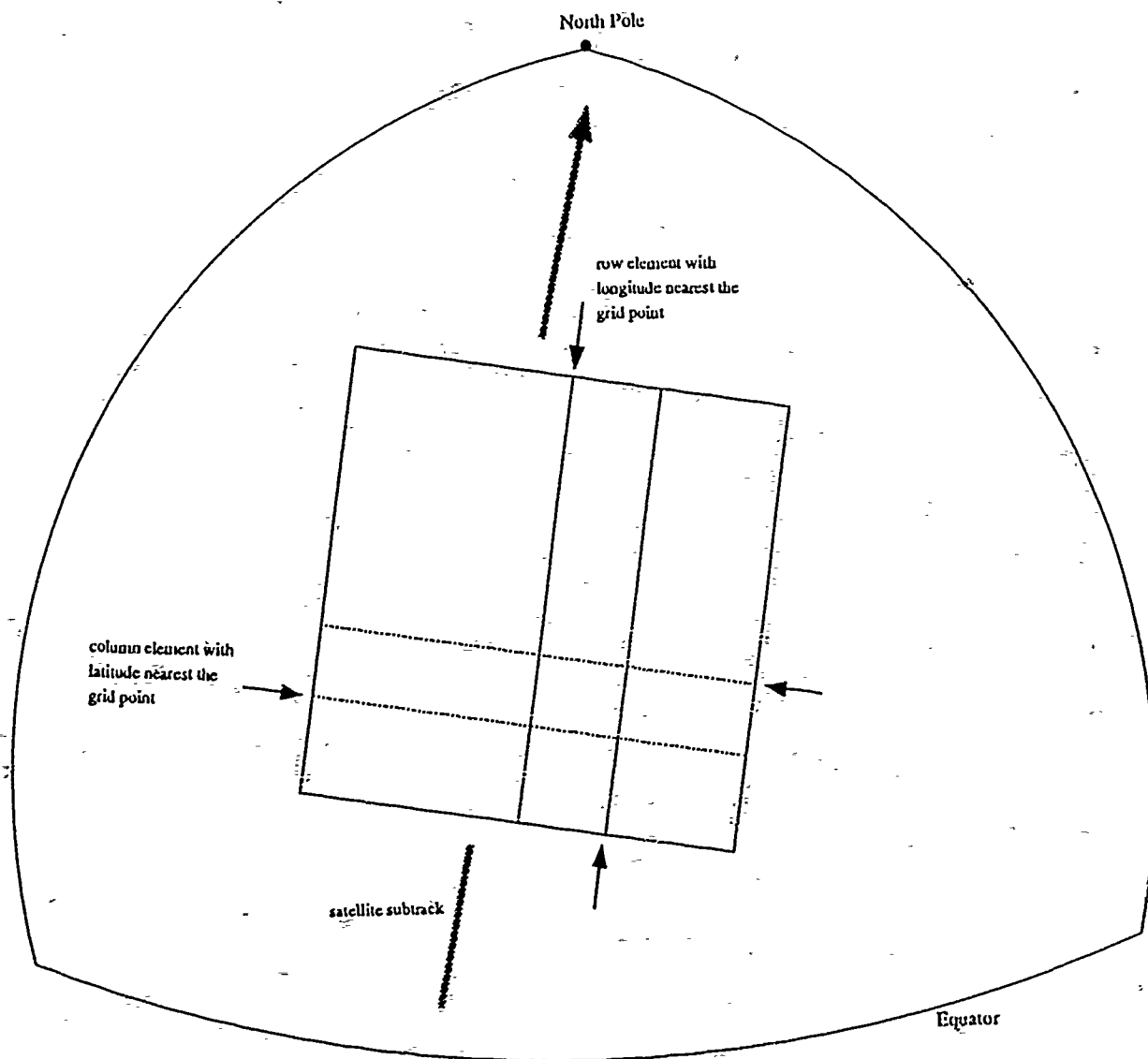


Figure 5.5 Binary Search Method. The large rectangle corresponds to a data array such that the data are oriented ascending/North up. The arrows mark the data latitudes and longitudes found by the search routine when it searches the first and last rows and columns. The small box in the center of the data array is the reduced array.

means that latitude and longitude change monotonically across the array rows and down the array columns. Note also that the data array is rectangular. The range of latitudes and longitudes in the array can be found by examining the corner points. As a result, in searching for the nearest neighbor to a grid point, it can be quickly determined whether the grid point lat/lon is included in the All_loc.dat array. If it is, a binary search of the array is done. Otherwise, no search is done. This criterion, based on the corner point information, eliminates useless searches. In some instances, the assumption of monotonicity is not valid and the binary technique cannot be used (see Section 5.4). In these cases, the alternative random search method is used.

5.4.1 The Binary Search Method

Prior to applying the binary search routine, the orientation of the data within the array must be determined. The orientation of the rows is related to the order in which the scan lines were stored on tape. If the first scan line is the northernmost line, the orientation is termed North up. Otherwise it is North down. The orientation of the columns is related to the direction in which the satellite was moving relative to the North Pole. This is characterized as either ascending or descending. When the satellite is ascending or moving towards the North Pole, in each scan line, the magnitudes of the latitude and longitude of the westernmost scan element are less than the magnitudes of the latitude and longitude of the easternmost scan element. When the satellite is descending or moving away from the North Pole, the magnitudes of the latitude and longitude of the westernmost scan element are greater than the magnitudes of the latitude and longitude of the easternmost scan element. The different column and row orientations result in the following combinations:

Ascending/North up	West Latitude < East Latitude West Longitude < East Longitude;
Ascending/North down	West Latitude > East Latitude West Longitude > East Longitude;
Descending/North up	West Latitude > East Latitude West Longitude < East Longitude; and
Descending/North down	West Latitude < East Latitude West Longitude > East Longitude.

An orientation-specific search routine is needed for each arrangement.

For example, assume a scenario such that the data are oriented Ascending/North up. Given a location grid, the binary search for the nearest neighbor to each grid point in the location grid works as follows. For a grid point which has been converted to lat/lon coordinates, a search is done for the grid point latitude along the westernmost column of the satellite location data array. The first and last elements, or corner points, in the column are examined. If the latitude falls between them, the column is bisected and the first and middle elements are examined. If the latitude is not contained there, then it must be contained between the middle and last elements. This bisection process continues in this column until the column element nearest to the grid point with respect to latitude is found. These steps are repeated for the latitude along the easternmost column and for the longitude along the northern and southernmost rows (See Figure 5.5). If at any time the initial check of the corner points in any row or column fails, the search is ended and no nearest neighbor is found since this implies the grid point lies outside the data array. It is important to note that this part of the binary search requires a rectangular data array and is confined to the first and last rows of the array.

If the initial search is successful, the location of the nearest neighbor point has been narrowed down to a smaller rectangular part of the data array. This reduced array is delimited by the row and column elements identified in the initial search. The binary technique is repeated on this reduced array using the new row and column boundaries as first and last rows. This process is repeated, narrowing the location of the nearest neighbor point each time until the reduced array contains only four points. The actual nearest neighbor of these four is determined using a random search. A standard Euclidean distance formula is used to compute the distance between the grid point and each of the four remaining data points. The data point with minimum distance is chosen as the nearest neighbor.

It is reasonable to assume that the nearest neighbors to adjacent grid points will be located in the same general area of the data array. The search for the next grid point begins on a new reduced data set located in the vicinity of where the previous point was found. If a nearest neighbor is not found in this reduced set, the whole data set is searched to ensure no mistakes occur.

5.4.2 The Random Search Method

In some instances, the location data does not change monotonically (See Section 5.6). The binary search method cannot be used to search arrays where the assumption of monotonicity fails to hold. In these cases, the random search is used. This technique

selects the data point with minimum Euclidean distance from the grid point as the nearest neighbor. A nearest neighbor is always found, regardless of how large the distance between the grid point and the data may actually be. This could result in incorrect data being mapped to the grid whenever the nearest neighbor is not geographically close to the grid point location. One way of avoiding these errors would be to place an upper limit on the distance between the grid point and the nearest neighbor. However under some circumstances, a fixed limit may result in data not being mapped that should be mapped. The magnitude of a Euclidean distance calculated from lat/lon information is not always indicative of the actual distance measured on the surface of the Earth. For example, close to the pole, large changes in longitude correspond to small changes in actual surface distance. A criteria similar to the corner point idea used in the binary search was developed to ensure spatial proximity between the grid point and the data. Analogous to using the corner points, the minimum and maximum latitudes and longitudes contained within the lat/lon array are used to decide whether or not the grid point lies within the location array. If the grid point falls within the range of data, a nearest neighbor is chosen. No limit is placed on the distance.

5.5 Mapping Considerations

The design of the mapping utility, in particular how the location grid points are processed, is determined in part by two things. The first involves the need for computational efficiency dictated by the number of grid points that must be processed and the large amount of data (See Section 5.1). The second involves the mapping of the satellite ephemeris data to the ephemeris grid according to the SGDB format.

Computational efficiency is a main objective in the design of the mapping utility. The two most time consuming parts are input of the AVHRR data and the search of the lat/lon array for nearest neighbors. If the number of times the data must be input and the number of searches are reduced, the overall efficiency will be improved. This is accomplished by making use of the following fact: the satellite orbits do not coincide with the Neph box structure; the scan lines cover portions of multiple Neph boxes as shown in Figure 5.1. Rather than processing the entire Neph grid row by row for each satellite orbit, the utility is constructed so that only those Neph boxes which will be partially or completely filled with data from the selected orbit are processed. Prior to running the mapping program, the Neph boxes which coincide with a specific orbit are determined. This is accomplished using a program which graphically displays the extent of the AVHRR satellite orbital data in scan line format. The scan line boundaries are superimposed on an RTNEPH polar

stereographic world map projection. Also displayed are the Neph box boundaries. From this, the Neph boxes to be processed for each orbit are chosen (see Figure 5.6).

The number of scan lines that can be processed in a single run is restricted. Due to large program memory requirements and computer system limitations, an upper limit on the amount of data that can be handled is imposed. A whole orbit worth of data exceeds this limit. The orbits must be divided into smaller data arrays of 500 scan lines. While allowing a reasonable amount of data to be processed, 500 was chosen because it did not result in problems with either system memory or excessive CPU time. Similarly, the number of Neph boxes processed per run is restricted to range from one to four. The maximum was set at four because, in general, the range of 500 lines of data will not extend beyond four Neph boxes.

The order in which the grid points are processed is also influenced by the criteria governing the construction of the ephemeris grid. Recall from Section 5.3 that according to the AFGWC SGDB standard, ephemeris data are maintained at quarter mesh resolution. Keeping with this standard, the sensor grids are constructed at 64th mesh resolution and the ephemeris grid at quarter mesh resolution.

A Neph box is divided into 1024 (32x32) quarter-mesh boxes each containing 256 (16x16) 64th mesh grid points. The 64th mesh grid points are processed row by row a quarter mesh box at a time. When a complete quarter mesh box has been processed, the sensor grids and the ephemeris grid are updated only if a nearest neighbor has been found for each of the 256 64th mesh points in the box. The ephemeris grid is updated with ephemeris data which has been degraded from 64th to quarter mesh by averaging over the 256 grid points. If fewer than 256 points are found, the grids are not updated.

5.6 Data Processing Considerations

As noted in the discussion of the binary and random search algorithms (see Section 5.4), certain attributes of the data cause problems in the selection and application of the search techniques. Some of these characteristics which occur consistently in the data, have been identified and are dealt with automatically in the program. Others are idiosyncrasies of specific data sets and must be handled individually.

Recall that the binary search routine used to find the nearest neighbor data points is built on the premise that in the AVHRR location data array, `All_loc.dat`, latitudes and longitudes change monotonically across the array rows and down the array columns. In the real satellite data, this assumption is valid at latitudes less than about 50 degrees North.

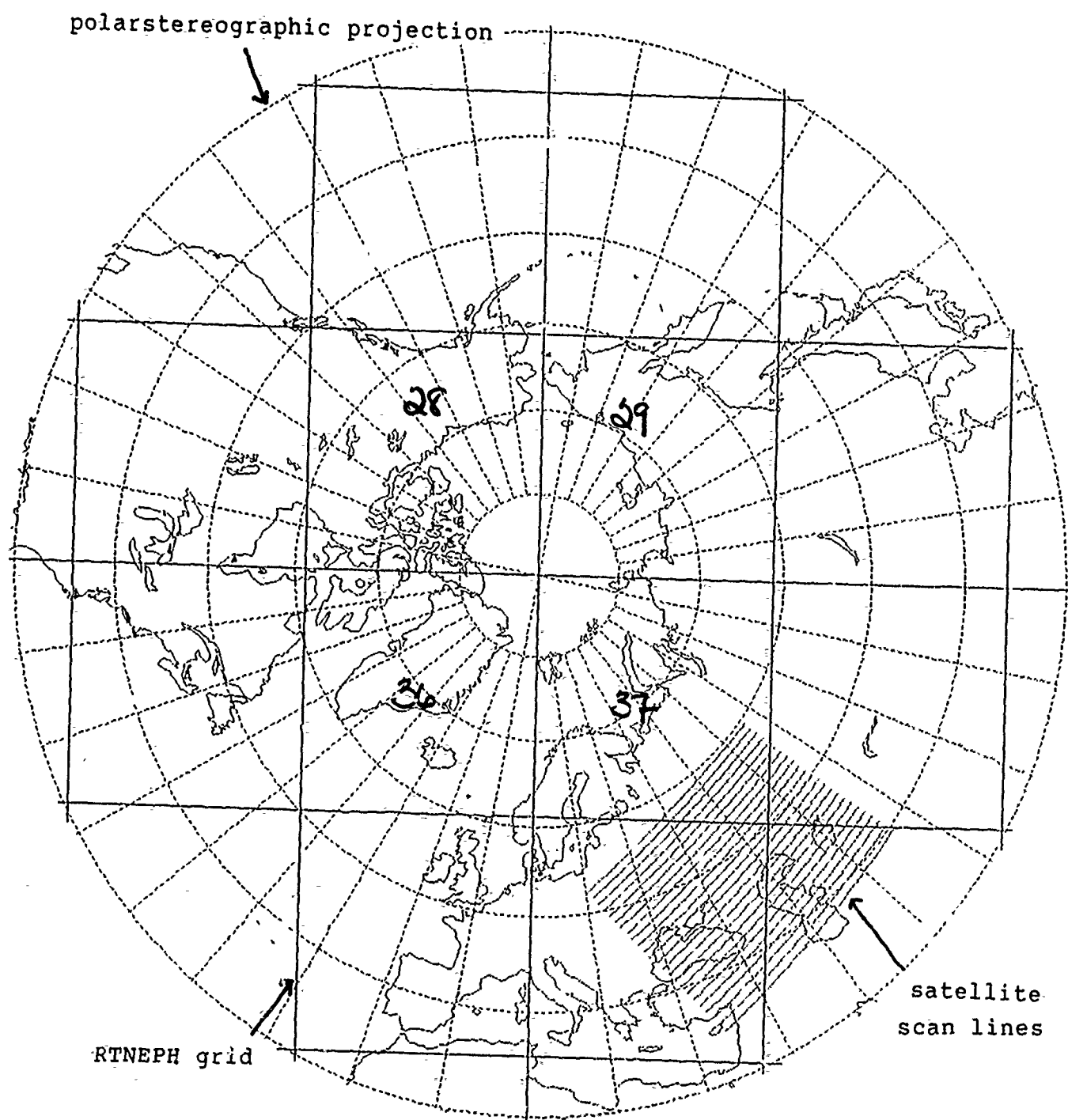


Figure 5.6 Schematic of AVHRR data in scanline format superimposed on a Northern Hemisphere Map projection. The RTNEPH grid is also displayed.

At higher latitudes, the location data are not always monotonic, either in latitude or in longitude, making use of the binary search method impossible. However, the circumstances under which this assumption fails are identifiable and modifications have been made which make the binary search applicable in most cases. The required modifications involve both the application of coordinate transformations to the AVHRR satellite location data and Neph grid structure and changes to the binary search routine. Three instances where the monotonicity assumption does not hold have been identified and are outlined below.

Two failures of the monotonicity assumption result directly from the definition of longitude. Because longitudes are defined to range from 0 to +180 degrees East and 0 to -180 degrees West, at 180 degrees longitudes shift instantaneously by 360 degrees. As a result, the longitudes of any scan line crossing 180 degrees do not change monotonically. For example, when the first and last scan elements of a scan line which crosses the 180 degree line are examined, the range of longitudes may appear to go from -160 to +170 degrees inclusive (monotonically increasing) when in reality the range is from -160 to -180 degrees (decreasing) and +180 to +170 degrees (decreasing). The two scenarios which occur are 1) the scan lines cross 180 degrees but do not approach the North pole (Figure 5.7a) and 2) the scan lines cross both the 180 degree line and the North Pole (Figure 5.7b). Note that this problem does not exist at the Greenwich meridian. Longitudes of scan lines crossing this meridian are simply centered around 0 degrees. In scenario 1 above, the solution to the problem caused by the 360 shift in longitude involves a transformation of the grid box structure and then a transformation of the AVHRR location data. Because the RTNEPH Cartesian grid is shifted 10 degrees off the Greenwich meridian (see Section 3.1.1), Neph boxes 25 through 28 cross 180 degrees and thus contain a 360 degrees shift (see Figure 5.1). The first step is to remove this shift by transforming the negative longitudes (-170 to -180 West) to positive ones (190 to 180) thereby making longitudes monotonically increasing within the boxes. A similar transformation is applied to the AVHRR location data to make the longitudes consistent. The base of the data is changed from 0 to +/- 180 degrees East or West to a +360 or -360 degree base. Which base to use is determined by whether the Neph boxes currently being processed are numbered between 1 and 32 or 33 and 64. If the boxes are numbered between 1 and 32, all the grid points will have longitudes between 10 degrees and 190 degrees. Negative longitudes are not involved, so the data are converted to a +360 degree base by adding 360 to all negative longitudes in the data set. Similarly, boxes 33 - 36 and 41 - 64 contain grid points with longitudes ranging from 0 to -170 degrees. Positive longitudes are not involved so the data are converted to a -360 degree base by subtracting

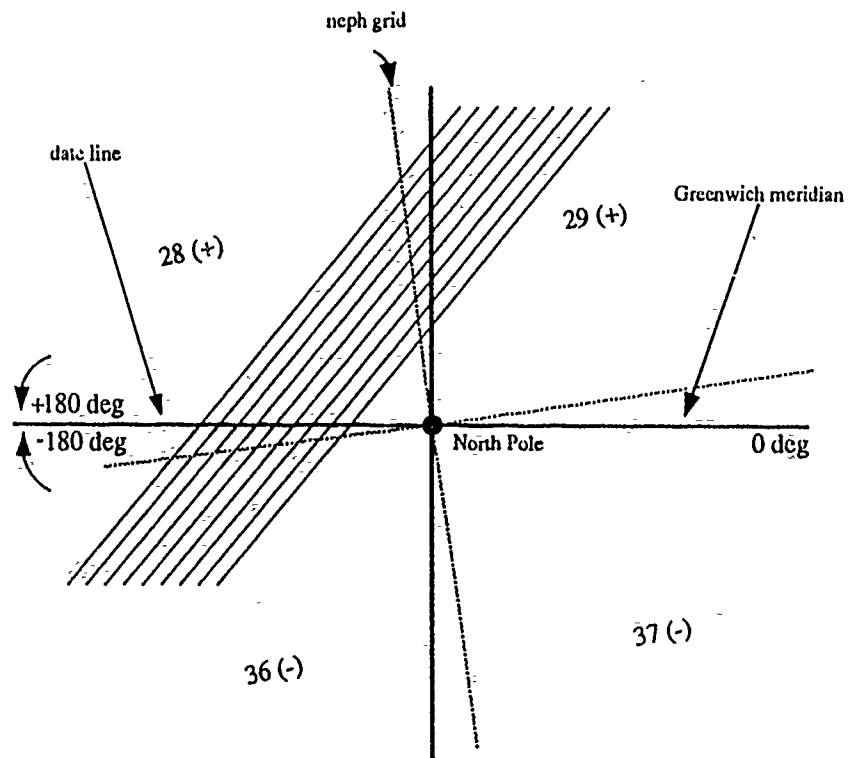


Figure 5.7a Examples of scan lines which cross only the dateline. The longitudes are converted to a + or - 360 deg base depending on which Neph box is being processed. Which base is designated by the (+) or (-) after the neph box number.

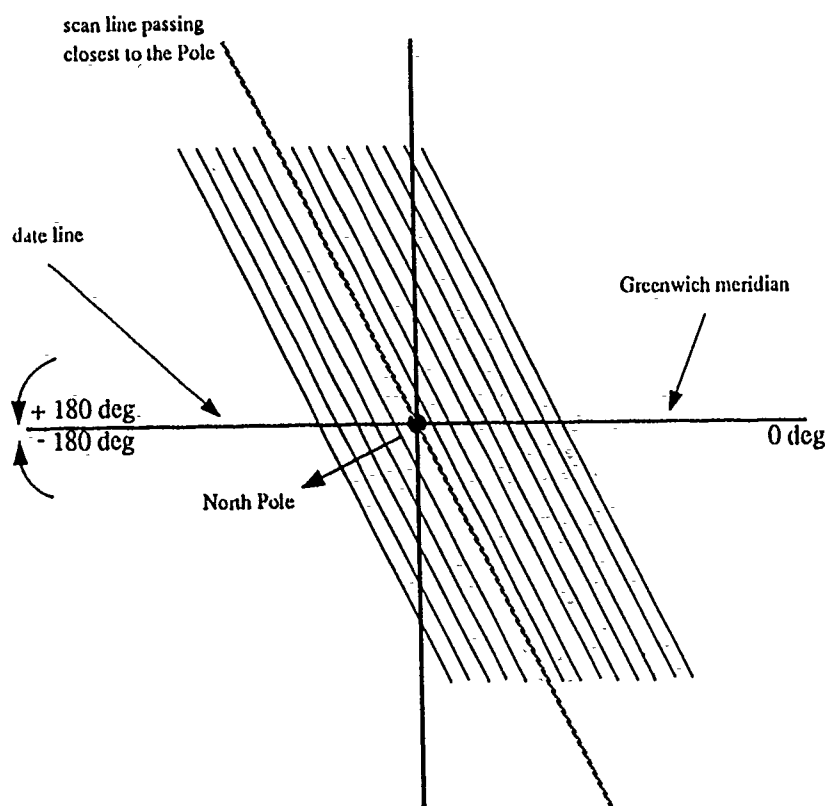


Figure 5.7b Example of scan lines which cross both the dateline and the pole. The data set is divided at the scan line nearest the pole. The latitudes of lines crossing the dateline are converted to a + or - 360 deg base as in Figure 5.7a. The other scan lines are left unchanged.

360 from all positive longitudes. Neph boxes 37 through 40 seemingly present a problem in that they contain both positive and negative longitudes in the area of the Greenwich meridian. However, no difficulty exists because it is geometrically impossible for a scan line to cross the 180 degree line and also lie partially within the positive part of the Neph box. This means the positive portion of the Neph box will not be filled by data from these scan lines so transformation of the data to a -360 degree base is required.

In scenario 2, where the scan lines cross both the 180 degree line and the pole, an additional problem exists making the solution more complicated. The added difficulty occurs when the scan lines cross the pole and the orientation of the data changes from ascending to descending (refer to Section 5.4). This switch in orientation coupled with the fact that only some of the scan lines cross the 180 degree line and contain a 360 degree shift is accommodated by dividing the data set into two parts at the scan line nearest the North Pole. Each part is treated as a separate data set. To find the scan line nearest the North Pole, the algorithm makes use of the fact that a scan line crossing the North pole will lie very nearly along a single line of longitude. By comparing the first and last scan elements of each line, the scan line lying nearest the Pole can be determined. The data set is divided at this line. One part contains scan lines which cross 180 degrees and is treated as in the first scenario. The other part is left unchanged since the search routine can handle the data as is.

The third failure of the monotonicity assumption occurs with respect to latitude and results from the curvature of the Earth at latitudes greater than 50 degrees North. For example, in a scan line at a high latitude the latitudes may initially be monotonically increasing, however, at a certain point a switch occurs and the latitudes decrease monotonically through the end of the scan line (see Figure 5.8).

This switch in monotonicity may not occur in every scan line but always occurs in successive scan lines. It also may not always occur in the same scan column. Recall the one to one correspondence between scan lines and data array rows. By partitioning the data array into rectangular subarrays as shown in Figure 5.9, it is possible to isolate the lines and columns of the data where the switches occur. In the subarrays where the data are monotonic the binary search technique is used, otherwise the random search is used. If the satellite data cross the North Pole, the data array is also divided at the Pole as described above.

If the random search subarray is located at the Pole, the data are not used because it is impossible to define whether the grid point lies within the subarray or not. For example, an incorrect point, (i.e., one on the wrong side of the pole) could easily be selected as the nearest neighbor since all longitudes are valid at the pole (see Section 5.4.2). However,

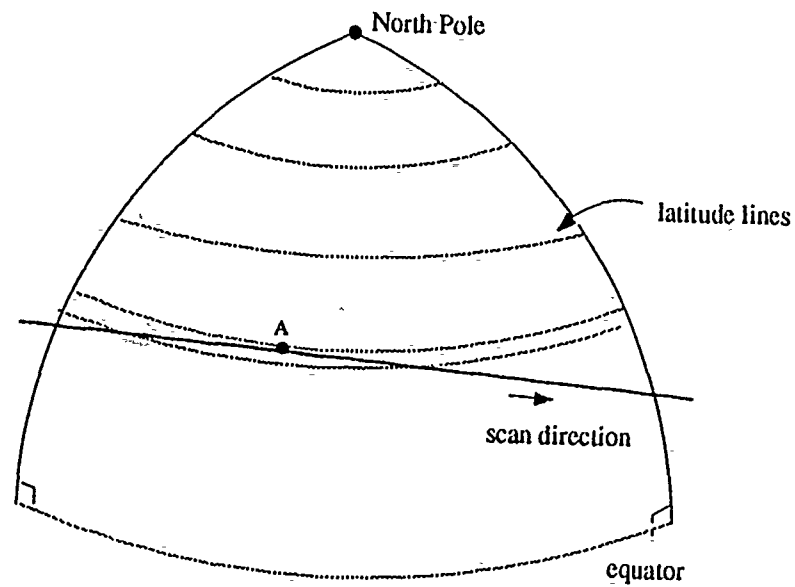


Figure 5.8 Switch in latitude trend within a scan line due to curvature of the Earth. To the left of point A, latitude is monotonically increasing along the scan line. To the right, latitude is monotonically decreasing.

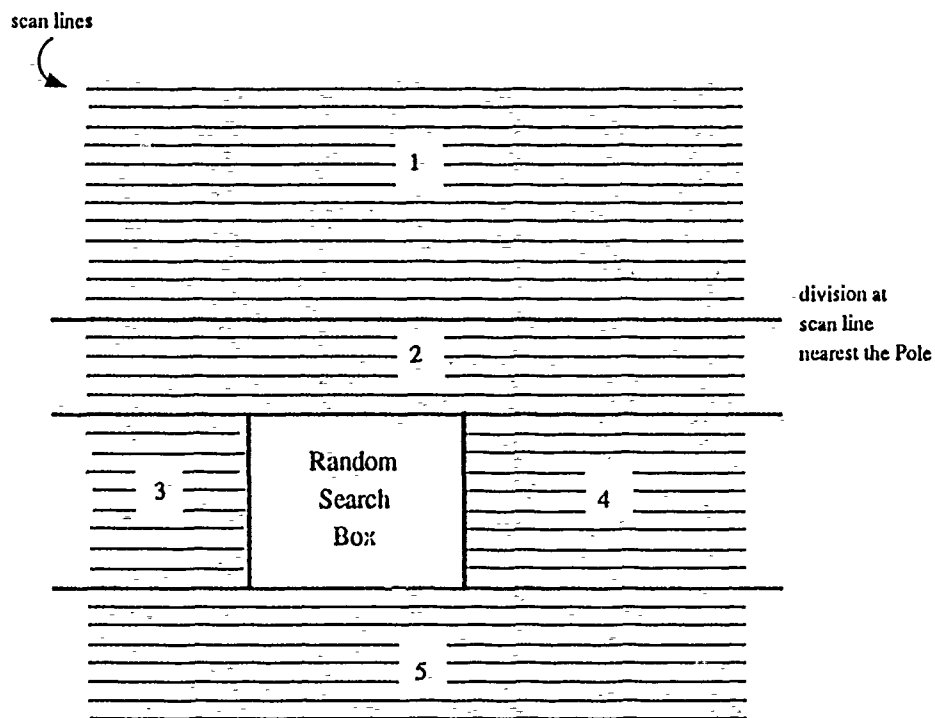


Figure 5.9 Division of data set into subarrays. The data are monotonic with respect to latitude within subarrays 1-5. The binary search technique is applied. The Random Search Box contains all non-monotonic data.

there are still sufficient data to create a complete polar stereographic map since each orbit in the entire tape data set contains data in the polar region.

A problem that must be handled on a per case basis involves scan lines which contain bad data or for which the satellite navigation routine has produced incorrect location information. These problems occur most often at the pole and at the equator. At the pole, navigation problems make a single scan line appear to spiral around the pole. At the equator, a scan line will appear to traverse the entire hemisphere. Another problem occurs when the satellite passes into the Southern Hemisphere. The data stream stops in the middle of a scan line and picks up again when the satellite reenters the Northern Hemisphere. Bad data such as these do not occur in each data set. Sometimes bad scan lines can be identified visually using the graphical display program used to view scan line extents. Other times it is simply necessary to exclude all scan lines near the pole or the equator from the data sets.

5.7 Unsolved Problems and Possible Solutions

Some problems exist which have been recognized but not solved. These are directly related to the AVHRR data set used in this project. Whether similar problems will occur with other data sets is not known. The first is an additional case where the monotonicity assumption fails; the longitudes are not monotonic down the scan columns. Where the other problems described above were predictable switches in scan line trend, in this case the longitudinal anomalies follow no apparent pattern. The problem, when it occurs, seems to be restricted to the first 30 columns of the AVHRR data set, as if the satellite sensor was vibrating at the beginning of the scan or the navigation routine wasn't working correctly. One possible solution is to check the data for serious inconsistencies and discard bad data. Unsolved, this problem results in the inability of the search routines to find nearest neighbors for some of the grid points, leaving holes in the map. Usually, the utility works past these problems. When it can't, the holes should be filled in by data from later orbits where scan coverage is duplicated.

A second problem is extremely long run times for certain combinations of satellite data from this data set and Neph boxes. For example, when processing the scan lines that pass over the pole from data tape #1 with Neph box 29, the mapping utility takes over 25 hours of CPU time to run to completion. In comparison, runs with different data on other Neph boxes can use as little as 15 minutes of CPU time. In this case, the binary search routine seems unable to quickly narrow the search area down to a small set of data. This

might be a result of the longitude problem mentioned above (i.e., an idiosyncrasy in the data that the search routine is unable to handle or bypass).

5.8 Summary

An AVHRR Mapping Utility was developed as part of the RDNEPH project at GL. It transforms satellite data from raw scan line format to the SGDB standard projection. This projection is based on the AFGWC RTNEPH polar stereographic grid system. In this development effort, AVHRR GAC data corresponding to nine quarter orbits over the Northern Hemisphere were reformatted to the RTNEPH grid. The result was a complete set of polar stereographic grids for all five AVHRR channels that are available for use in the RDNEPH.

The mapping algorithm relies on the latitude/longitude information associated with both the grid and the satellite data. For every grid point in lat/lon coordinates, a search is done of the satellite location data to find the data point spatially closest to the grid point. Sensor and ephemeris data associated with the nearest neighbor point are mapped into the grid. Two different search methods, a binary search and a random search are used to find the nearest neighbor. The preferred method is the fast efficient binary search, however, it is not always applicable due to certain characteristics of the satellite data. Modifications to both the search algorithm and the data make the binary search technique usable in most cases. The random search is used only when the binary algorithm cannot be applied.

The large amount of data that must be mapped requires that the mapping process is computationally efficient. The algorithm design and processing methodology result directly from this requirement. Only those Neph grid boxes for which satellite data actually exist are processed.

Some problems were encountered for which no solutions were discovered. These problems are seemingly the result of idiosyncrasies of specific data sets. Future work may include a more careful analysis of the problems and a search for solutions.

6. A New Procedure for the RTNEPH Infrared Water Vapor Attenuation Correction

6.1 Background

The RTNEPH infrared processor generates a cloud analysis based on satellite data that is measured in the 11 μm window. Cloud/no-cloud decision and cloud-top height estimation are performed in this module by comparing a satellite-measured brightness temperature to estimates of the actual surface skin temperature and upper-air temperature profile valid at the location being analyzed. In RTNEPH, clouds are treated as black-bodies and it is assumed that temperature decreases monotonically with height throughout the troposphere. Under such circumstances, a particular scene will be considered cloud-free if the measured brightness temperature is greater or equal to the true ground temperature; if it is lower, the scene is determined to contain clouds. The altitude of the atmospheric level at which reference air temperature equals the observed brightness temperature is taken as an estimate of the actual cloud-top height.

The 11 μm window is a spectral region in which the atmosphere is nearly transparent to radiation emitted by the earth's surface. However, the effects of water vapor attenuation may become significant in the tropics and for high satellite viewing angles. For instance, a moist atmosphere may attenuate the lower tropospheric emission by 5 to 10%, causing the surface being viewed to appear as much as 7-8 K colder than it truly is. One obvious consequence is that if a comparison is made between the uncorrected satellite measurement of brightness temperature and the actual surface skin temperature, the infrared processor may detect the presence of clouds when in fact the atmosphere is cloud free. It is therefore important to properly account for the effects of the water vapor attenuation on the IR brightness temperature measurements prior to performing the cloud analysis. In the current version of RTNEPH, several empirical corrections are applied to the raw satellite data which help compensate for the effects of the water vapor attenuation. These complicated corrections are functions of such parameters as the viewing geometry and the latitude of the observation, but do not in general depend on the actual atmospheric composition. In order to remedy the deficiencies of the current approach, it was suggested to base the water vapor attenuation calculations on radiative transfer modeling, using humidity profiles obtained from operational forecast or analysis models. The magnitude of the brightness temperature correction cannot be determined exactly with such an approach, owing to the uncertainties on the model parameters and on the moisture and temperature data. However, it is reason-

able to expect that the use of a physically rigorous method of calculating the water vapor attenuation correction will have a positive impact on the quality of the RTNEPH analysis.

One minor conceptual problem arises in the implementation of the proposed method for application to cloud-top height estimation. In this case, the amount of absorber in the atmosphere above the cloud is not known a priori, making it impossible to evaluate the correction on the observed brightness temperature. The problem can be solved in practice by modifying the upper-air reference temperature data to include the effects of the water vapor attenuation. In this way, cloud-top height can be determined by direct comparison of the uncorrected satellite measurement with the new temperature profile.

In the first phase of our study we had to develop a radiative transfer model which meets both accuracy and computational requirements for application to the RTNEPH cloud analysis. The model proposed by Weinreb and Hill (1980) for the simulation of NOAA's NESS satellite radiometers which operate in the infrared windows seemed suitable for RTNEPH purposes at least at an experimental stage. Section 6.2 describes a fast algorithm for the calculation of the water vapor attenuation correction which incorporates Weinreb and Hill's formalism. The scheme was designed to approximate the AFGL FASCOD line-by-line calculations (Clough et al., 1981) which we used here as a reference. The radiative transfer algorithm was subsequently incorporated to RTNEPH (see section 6.3). The positive impact of this new procedure on the cloud analysis is demonstrated in section 6.4.

6.2 Radiance Calculations

6.2.1 Radiative Transfer Theory

In the radiative transfer calculations, the atmosphere is represented by a sequence of L homogeneous layers, each layer ℓ being assigned a mean temperature T_ℓ and a mean pressure P_ℓ . By convention, the layers are numbered starting at the top of the atmosphere. The temperature at the lower boundary of layer ℓ (ℓ th level) will be denoted by T_ℓ^* .

The upwelling radiance, I_ν , at wavenumber ν , measured at the top of an atmosphere within which lies a blackbody at level ℓ , can be numerically evaluated as:

$$I_\nu = B_\nu(T_\ell^*)\tau_{\nu,\ell} + \sum_{i=1}^{\ell} B_\nu(T_i)(\tau_{\nu,i-1} - \tau_{\nu,i}). \quad (1)$$

where $B_\nu(T)$ is the Planck emission for a blackbody at temperature T , and $\tau_{\nu,i}$ is the spectral transmittance between the top of the atmosphere and the i 'th level.

Equation (1) holds only for monochromatic radiation. Satellite sensors detect radiant energy only over some finite wavenumber interval $[\nu_1, \nu_2]$. The total radiance I measured by the satellite radiometer is obtained by convolving the spectral radiance I_ν with the instrumental response function ϕ_ν ,

$$I = \frac{\int_{\nu_1}^{\nu_2} I_\nu \phi_\nu d\nu}{\int_{\nu_1}^{\nu_2} \phi_\nu d\nu}. \quad (2)$$

The response function of the DMSP/OLS-T sensor, the primary sensor used in RTNEPH, is shown in Figure 6.1.

For the present application, it is preferable to work with equivalent brightness temperatures rather than radiances. The atmospheric radiances are converted to brightness temperatures through a look-up table. The table contains blackbody radiances $I(T)$ computed from equation (2) with I_ν replaced by $B_\nu(T)$, for temperatures T ranging from 190 K to 330 K.

6.2.2 Fast Frequency Integration

In the 11 μm window, the water vapor absorption varies on a scale of the order of 0.01 cm^{-1} . Therefore, in order to numerically evaluate the integral in equation (2), one must calculate I_ν for a large number of closely spaced values of ν . This approach is too computationally intensive for real applications; from this point of view it is far more advantageous to deal with spectrally-averaged water vapor absorption properties.

A fast procedure for performing the integration in equation (2) is based on the fact that the rapid spectral variations of the water vapor absorption are essentially uncorrelated with the Planck function and the sensor's response function f_μ . Therefore, if we choose a subinterval Du small enough so that the characteristics of the water vapor absorption (e.g., mean line strength, mean line spacing) do not change appreciably within $\Delta\nu$, then the following approximation is valid:

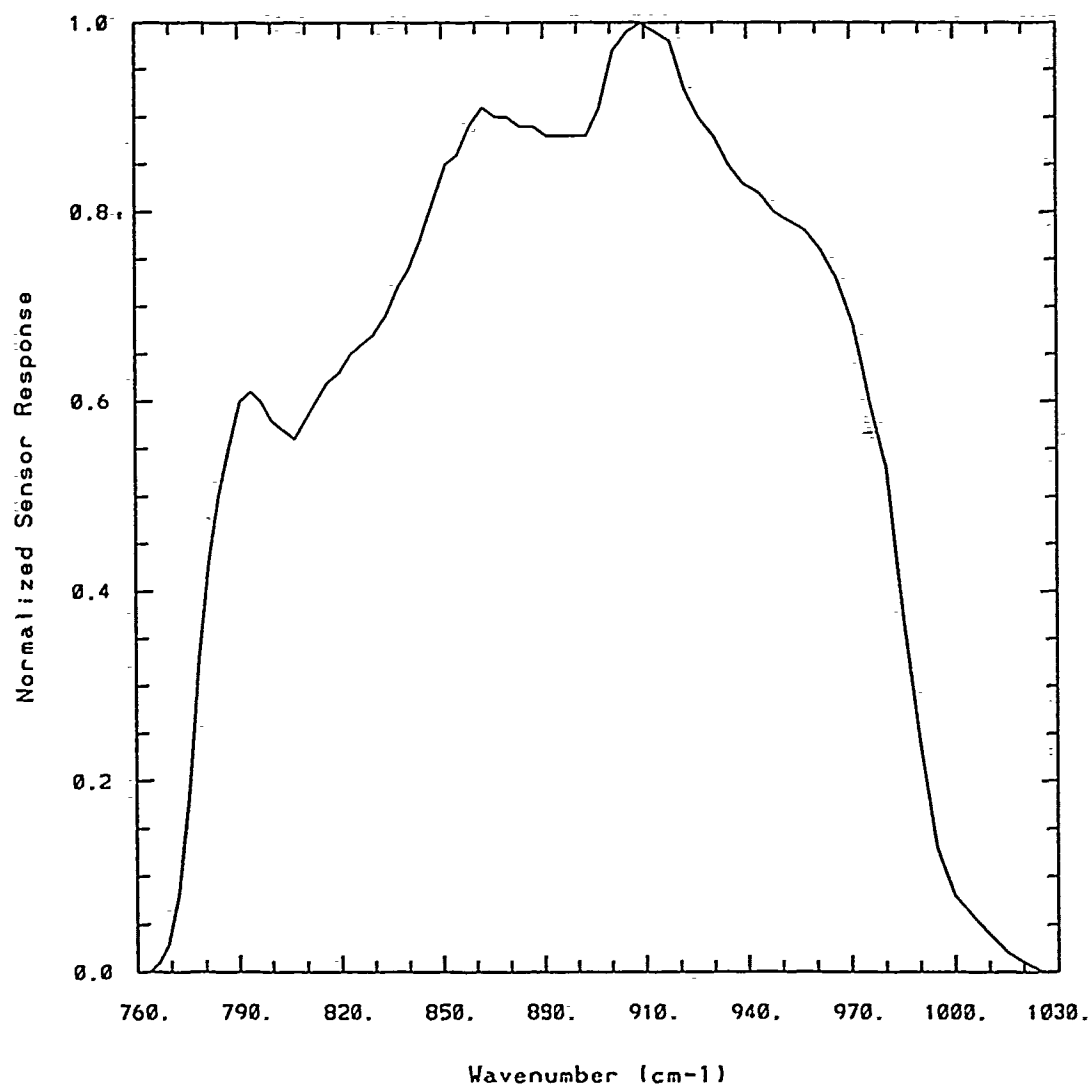


Figure 6.1 Relative spectral response of the OLS-T sensor.

$$\int_{\Delta v} \tau_{v,i} B_v(T_i) \phi_v dv \cong \frac{1}{\Delta v} \int_{\Delta v} \tau_{v,i} dv \cdot \int_{\Delta v} B_v(T_i) \phi_v dv. \quad (3)$$

In general, the above assumption does not hold over the full instrument bandwidth. In this case, the interval $[v_1, v_2]$ is broken into M subintervals Δv_m , $m = 1, \dots, M$, and approximation (3) is applied to each subinterval separately. By combining equations (1) and (2) and applying this procedure, we obtain the following approximate expression for the satellite-measured radiance I :

$$I \cong \sum_{m=1}^M \bar{\tau}_{m,\ell} f_m(T^*_{\ell}) + \sum_{m=1}^M \sum_{i=1}^{\ell} f_m(T_i) (\bar{\tau}_{m,i-1} - \bar{\tau}_{m,i}) \quad (4)$$

where

$$\bar{\tau}_{m,i} = \frac{1}{\Delta v_m} \int_{\Delta v_m} \tau_{v,i} dv \quad (5)$$

is the spectrally-averaged water vapor transmittance for which an efficient parameterization can be found, and

$$f_m(T) = \frac{\int_{\Delta v_m} B_v(T) \phi_v dv}{\int_{v_1}^{v_2} \phi_v dv}. \quad (6)$$

The spectrally-averaged Planck radiance $f_m(T)$ can be pre-computed and tabulated for an ensemble of values of T . One of the advantages of the above formulation is that the atmospheric transmittance τ is decoupled from the instrument response function ϕ . Consequently, the same parameterization of $\bar{\tau}_m$ can be used to simulate sensors of different spectral characteristics.

The results of a series of experiments involving various realistic response functions tend to indicate that, in the $11 \mu\text{m}$ window, the precision of the calculated brightness temperature when approximation (4) is used remains within 0.02 K for $\Delta v \leq 60 \text{ cm}^{-1}$.

In their simulation study, Weinreb and Hill (1980) chose to divide the 760-1000 cm^{-1} interval into 8 subintervals, each subinterval being 30 cm^{-1} in width. For reasons that are made clear in the next section, we decided to adopt the same frequency partitioning. However, the spectral interval had to be extended to 760-1030 cm^{-1} to cover the DMSP OLS-T region entirely.

6.2.3 Transmittance Modeling

In each subinterval $\Delta\nu_m$, the average water vapor transmission $\bar{\tau}_m$ is treated as a product of the transmittances of spectral lines and of self and foreign broadened continua,

$$\bar{\tau}_m = \bar{\tau}_m^{\text{lines}} \cdot \bar{\tau}_m^{\text{self}} \cdot \bar{\tau}_m^{\text{for}}. \quad (7)$$

This formulation is obtained by neglecting the spectral variations of the continuum components within $\Delta\nu_m$, which is consistent with the constraints imposed previously for the choice of the width of the subinterval.

6.2.3.1 Water Vapor Lines

The monochromatic transmittance of the water vapor lines along a given path through a layered atmosphere is

$$\tau_{\nu, \ell} = e^{-\sum_{i=1}^{\ell} \kappa_{\nu}(P_i, T_i) \rho_{\omega} \Delta s_i} \quad (8)$$

where κ_{ν} is the absorption cross-section at wavenumber ν ($\text{cm}^2/\text{mol.}$), ρ_{ω} is the number density of the water vapor (mol./cm^3) and Δs_i is the length in cm of the geometric path through layer i .

The problem of parameterizing the atmospheric transmittance of spectral lines, averaged over a given finite subinterval $\Delta\nu$, is largely complicated by the pressure and temperature dependence of the absorption coefficient κ_{ν} . Although they cannot lead to an exact mathematical treatment of the effects of the pressure and temperature variations along an atmospheric path, most methods employ narrow band transmission functions defined for fixed pressure and temperature as

$$\tau_{\Delta\nu}(P, T; u) = \frac{1}{\Delta\nu} \int_{\Delta\nu} e^{-\kappa_\nu(P, T)u} d\nu \quad (9)$$

where the variable $u = \rho\omega s$ represents the optical path length measured along the homogeneous path. Such functions are given in practice in the form of an analytical expression or as an ensemble of tabulated values.

The algorithm proposed by Weinreb and Neuendorffer (1973) is an approximate iterative procedure for calculating atmospheric profiles of transmittance, assuming that a model for the homogeneous path transmittances $\tau_{\Delta\nu}(P, T; u)$ is provided. One advantage of this procedure is that it requires no empirical parameter tuning and can therefore be applied in a straightforward manner to any atmospheric profile and any spectral domain, with the restriction that the technique works better for downward paths. Following Weinreb and Neuendorffer's procedure, the transmission $\bar{\tau}_\ell$ between the top of atmosphere and the ℓ^{th} level is calculated as

$$\bar{\tau}_\ell = \tau_{\Delta\nu}(P_\ell, T_\ell; u^*), \quad (10)$$

with

$$u^* = u_\ell + w \quad (11)$$

where u_ℓ is the amount of absorber in layer ℓ and w is determined using

$$\tau_{\Delta\nu}(P_\ell, T_\ell; w) = \bar{\tau}_{\ell-1}. \quad (12)$$

As suggested by the same authors, the following polynomial representation may be used for $\tau_{\Delta\nu}(P, T; u)$,

$$\ln(-\ln \tau_{\Delta\nu}) = \sum_{i=1}^{14} C_i(v) X_i \quad (13)$$

where $X_1 = 1$ $X_2 = 0.1 \ell n \left(\frac{uT}{273} \right)$ $X_3 = \ell n \frac{P}{1000}$

$X_4 = \ell n \frac{T}{273}$ $X_5 = X_2 X_3$ $X_6 = X_2 X_4$

$X_7 = X_2^2$ $X_8 = X_4 X_7$ $X_9 = X_3 X_4$

$X_{10} = X_2 X_7$ $X_{11} = X_4 X_6$ $X_{12} = X_4^2$

$X_{13} = X_3 X_6$ $X_{14} = X_3 X_7$

The coefficients $C_i(v)$ are determined by a least-squares fitting of equation (13) to line-by-line calculations of transmittance for a large number of homogeneous paths (between 100 and 200, typically). The conditions T , P and u for all paths must be carefully selected to correspond to the range of conditions encountered in real atmospheres.

The Weinreb and Nuendorffer's algorithm has been implemented on the AIMS system and tested. In our case, the coefficients $C_i(v)$ for the first 8 subintervals covering the region between 760 and 1000 cm^{-1} were taken from Weinreb and Hill's study (1980). The set of coefficients for the 9th subinterval, 1000 - 1030 cm^{-1} , was derived using the procedure described previously. The polynomial coefficients for the 9 subintervals are listed in Table 6-1.

Table 6-1 Polynomial coefficients for spectral lines of water vapor for the nine 30 cm^{-1} subintervals covering the 760-1030 cm^{-1} spectral region.

SubInterval	1	2	3	4	5	6	7	8	9
From(cm^{-1})	760	790	820	850	880	910	940	970	1000
To(cm^{-1})	790	820	850	880	910	940	970	1000	1030
$C(v,1)$	-2.23135	-1.85030	-3.09180	-2.81430	-3.27104	-3.87608	-3.96672	-4.05978	-3.61966
$C(v,2)$	6.12346	5.82894	7.02594	6.57833	7.08477	7.84570	7.95368	8.80584	8.10612
$C(v,3)$	0.45051	0.41698	0.32994	0.39086	0.36451	0.28824	0.26595	0.18746	0.19110
$C(v,4)$	2.71498	2.97656	3.39958	3.97101	3.99316	5.04843	4.60590	5.10449	2.97343
$C(v,5)$	1.20255	1.11404	1.23004	1.26900	1.13824	1.23165	1.02471	1.06154	0.41864
$C(v,6)$	-1.96786	-2.67401	-3.45263	-4.46690	-2.29048	-2.31676	-1.36628	-2.46299	-6.68960
$C(v,7)$	-3.76620	-3.92641	-6.40380	-5.74839	-5.69290	-5.89552	-4.72846	-5.15066	-4.43777
$C(v,8)$	0.16593	-2.77498	-1.00006	-0.50223	-0.64269	-3.71300	-3.68452	-9.38148	-0.11294
$C(v,9)$	0.43179	0.31976	0.61283	0.78782	0.30803	0.36473	0.30339	0.43482	-3.16484
$C(v,10)$	3.83257	2.79153	5.99156	6.60689	7.74251	6.08009	5.13370	2.28616	-1.78329
$C(v,11)$	2.18960	0.32945	0.66930	0.48105	-2.16183	-3.61897	-1.26021	0.52204	39.00848
$C(v,12)$	-1.57369	-1.52147	-0.59443	-2.76763	-0.78133	-0.21804	-0.67621	-2.57286	14.46646
$C(v,13)$	3.31522	3.42676	7.08475	7.00733	3.28952	6.03967	5.42607	6.41268	-7.10123
$C(v,14)$	-1.34526	-1.05728	-4.82659	-4.61433	-5.12407	-4.22350	-4.19546	-3.49313	-0.09407

Profiles of transmittances computed for the 760-1030 cm^{-1} interval with Weinreb's method have been compared to corresponding FASCODE line-by-line calculations. Figure 6.2 shows an example of the results obtained for a tropical atmosphere.

6.2.3.2 Water Vapor Continuum

We treat the water vapor continuum absorption in the same way as it is treated in the FASCODE line-by-line model (see Clough et al., 1981). At a given wavenumber, the total absorption coefficient due to self and foreign broadened continuum, $\kappa_c(\nu)$ ($\text{cm}^2/\text{mol.}$) is determined as,

$$\kappa_c(\nu) = \nu \tanh\left(\frac{\beta\nu}{2}\right) \cdot \left\{ \frac{\rho_s}{\rho_o} \bar{C}_s(\nu, T) + \frac{\rho_f}{\rho_o} \bar{C}_f(\nu, T) \right\}$$

where $\beta = hc/kT$ (in cm), ρ_s and ρ_f are the number densities in mol./cm^3 of the water vapor and the broadening gases respectively, and $\bar{C}_s$ and $\bar{C}_f$ in $\text{cm}^2/(\text{mol. cm}^{-1})$ are wavenumber dependent continuum absorption parameters for the self and foreign components. The quantity ρ_o is a reference number density at $P_o = 1013 \text{ mb}$ and $T_o = 296 \text{ K}$.

The strong temperature dependence of the self-broadened component is treated by storing the values of $\bar{C}_s(\nu, T)$ for $T=260 \text{ K}$ and 296 K , and by exponentially interpolating between these two sets of values. According to Clough et al. (1989), the temperature dependence of $\bar{C}_f$ can be ignored.

The values of $\bar{C}_s$ and $\bar{C}_f$ used in our model are averages over 30 cm^{-1} wide spectral intervals of the values provided in FASCODE. The numbers obtained for the 760 - 1030 cm^{-1} region are listed in Table 6-2.

Table 6-2 Coefficients for self and foreign broadened water vapor continuum for the nine 30 cm^{-1} subintervals covering the 760-1030 cm^{-1} region.

Subinterval	$C_s(\nu, 296)$	$C_s(\nu, 260)$	$C_f(\nu)$
760- 790 cm^{-1}	6.0983781E-05	1.2829671E-04	5.7394988E-08
790- 820 cm^{-1}	4.8957070E-05	1.0379052E-04	4.6192262E-08
820- 850 cm^{-1}	3.9661027E-05	8.4392414E-05	3.5695756E-08
850- 880 cm^{-1}	3.2440686E-05	6.9010923E-05	3.0600763E-08
880- 910 cm^{-1}	2.6788633E-05	5.6779423E-05	2.7922260E-08
910- 940 cm^{-1}	2.2337339E-05	4.7046633E-05	2.6445692E-08
940- 970 cm^{-1}	1.8830116E-05	3.9336694E-05	2.5705789E-08
970-1000 cm^{-1}	1.6293050E-05	3.3305692E-05	2.5330991E-08
1000-1030 cm^{-1}	1.4006365E-05	2.8697270E-05	2.5285331E-08

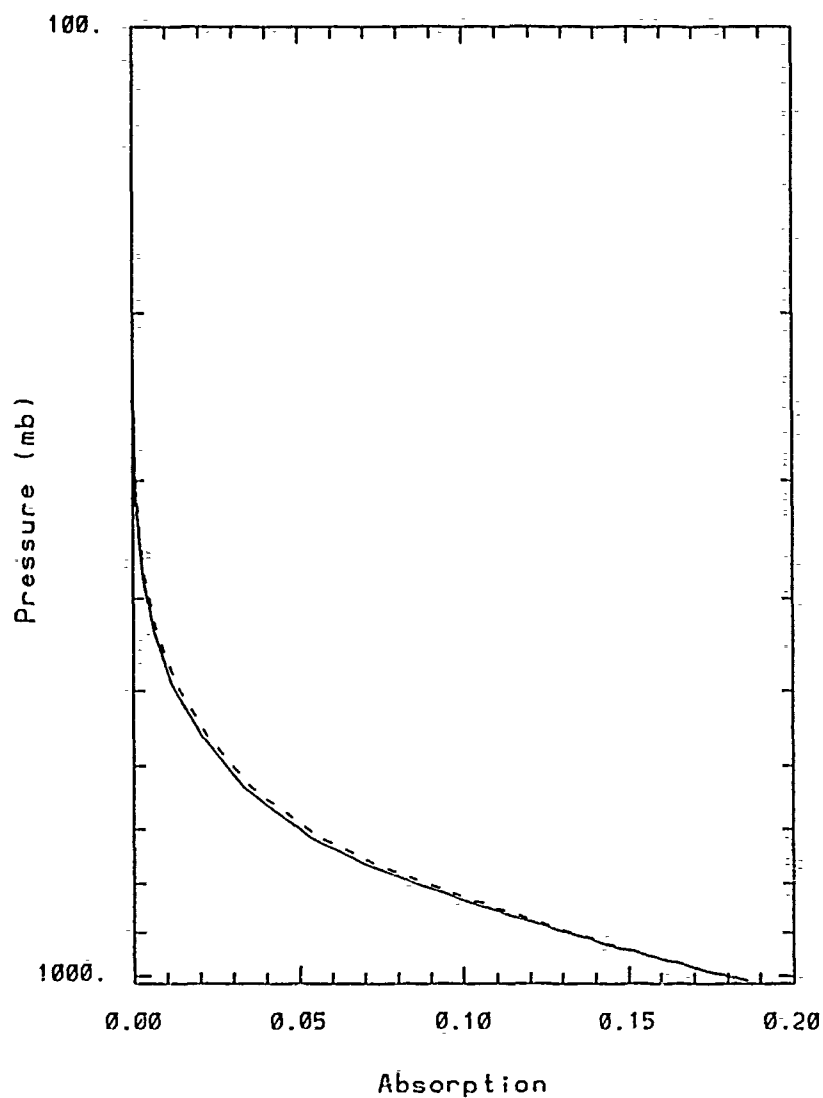


Figure 6.2 Comparison of water vapor absorption profile in the 760-1030 cm^{-1} region obtained with Fascode (—) and the fast algorithm (---) for a standard tropical atmosphere and a 60 degree zenith angle. Continuum absorption is not included in this case.

6.2.4 Comparisons with Line-by-Line Calculations

The rapid radiative transfer algorithm described previously is primarily intended to be used in RTNEPH for simulating measurements of atmospheric brightness temperatures by the DMSP/OLS-T sensor. In this section, the impact of the transmittance errors on the simulated satellite brightness temperatures is assessed based on comparisons with FASCODE line-by-line calculations.

Radiative transfer calculations have been performed with both models for each of the six AFGL standard atmospheres (see Anderson et al., 1986) and for two extreme values of the viewing angle, 0 and 60 degrees, in order to simulate the wide range of conditions normally encountered in the present application. For each profile, the top of the atmosphere was chosen to coincide with the troposphere. While neglecting the stratospheric water vapor has negligible impact on the results in the 11 μm window, it allows to speed up considerably the FASCODE calculations. The lower atmospheric boundary may coincide with the earth's surface as well as cloud tops and therefore might occur at any altitude within the troposphere. These various possibilities were accounted for by performing successive runs of the radiative transfer algorithms with the lower boundary displaced upward by 1-km increments between each run. The resulting profiles of atmospheric attenuation are shown in Figures 6.3, 6.4 and 6.5. Here, the term "atmospheric attenuation" refers to the difference between the true temperature of the surface being viewed (taken to be the ambient air temperature) and the brightness temperature measured at the top of the atmosphere.

From inspection of these figures, we see that the magnitude of the error on the attenuation calculations tends to be larger near the ground and decreases steadily for increasing altitudes. In any case, the discrepancy between the two models never exceeds 0.1 K which is much less than the thermal resolution of an 8-bit infrared satellite data base (approximately 0.5K).

6.3 RTNEPH Implementation and Timing Estimates

At the current stage of implementation, the radiative transfer calculations are being used for the sole purpose of determining the total atmospheric attenuation of the ground emission in support of the cloud/no-cloud decision process. Application to cloud height estimation is not yet supported.

The water vapor attenuation correction scheme is designed to accept upper-air temperature and humidity fields from operational forecast or analysis models. It is assumed that the data are mapped on a whole-mesh grid. The scheme expects estimates of

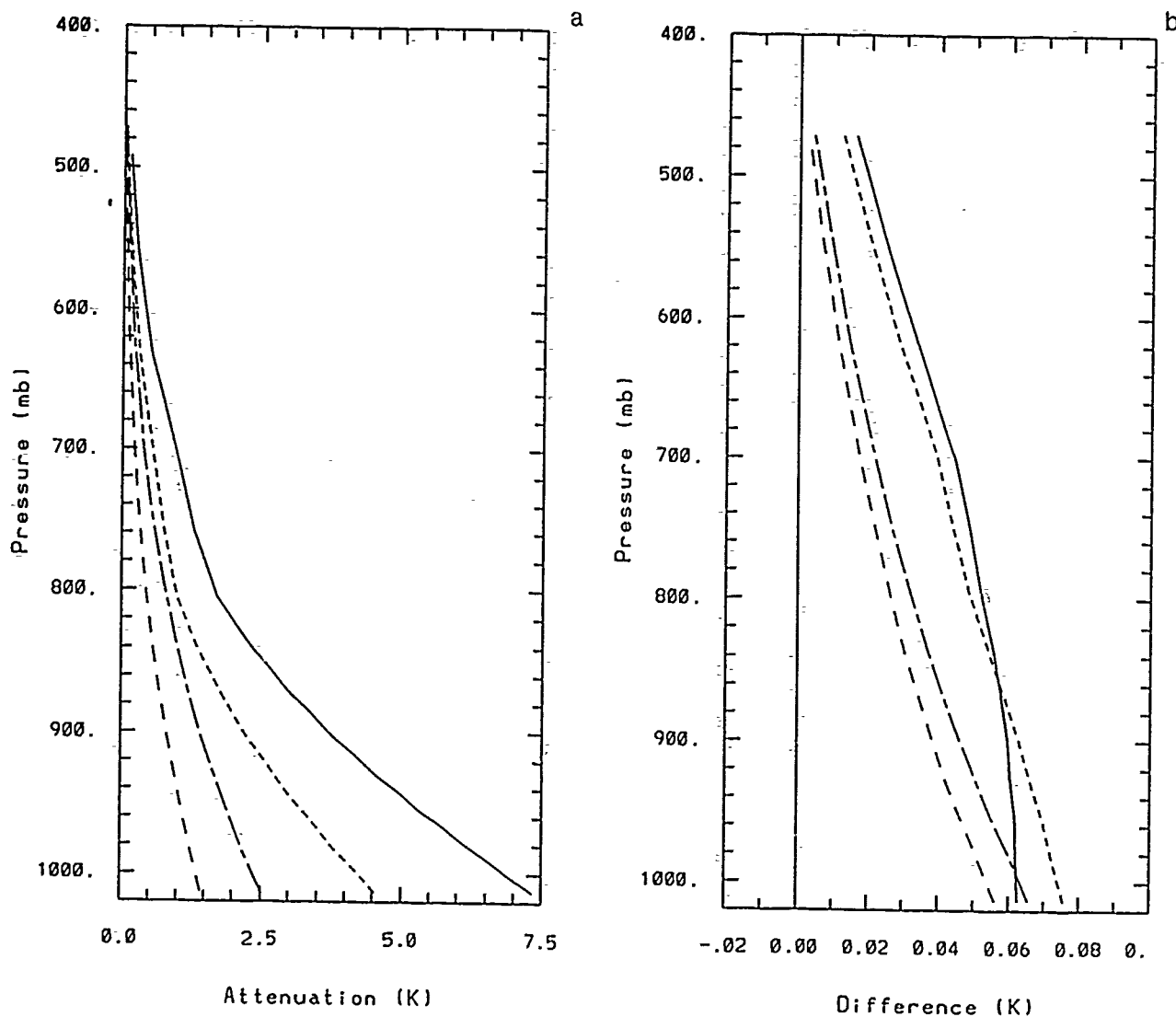


Figure 6.3 Water vapor attenuation profiles in the DMSP/OLS-T band obtained with Fascode for different atmospheres and different values of the viewing angle (a). The corresponding differences between approximate calculations and Fascode results are plotted in (b).

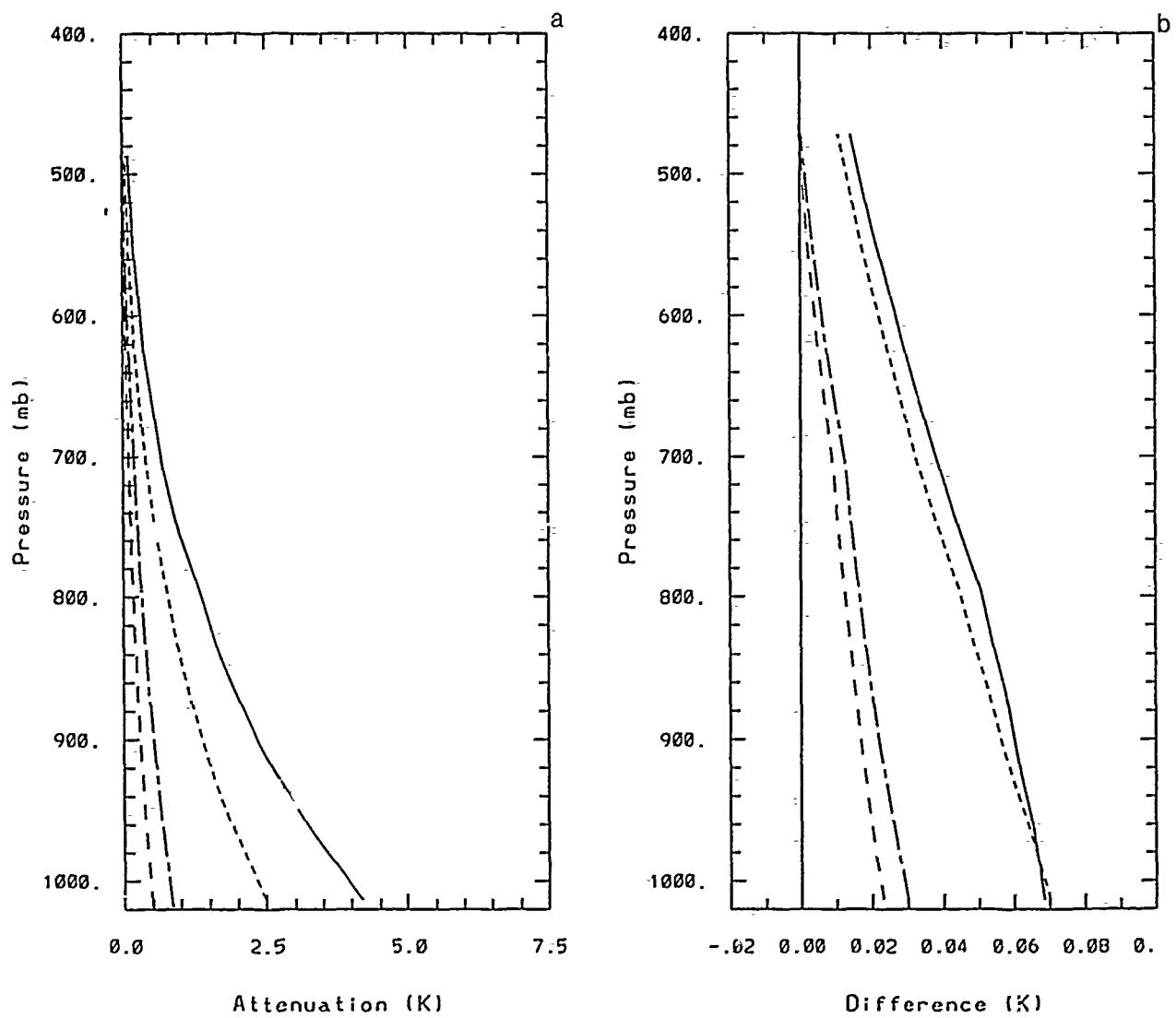


Figure 6.4 Same as Figure 6.3.

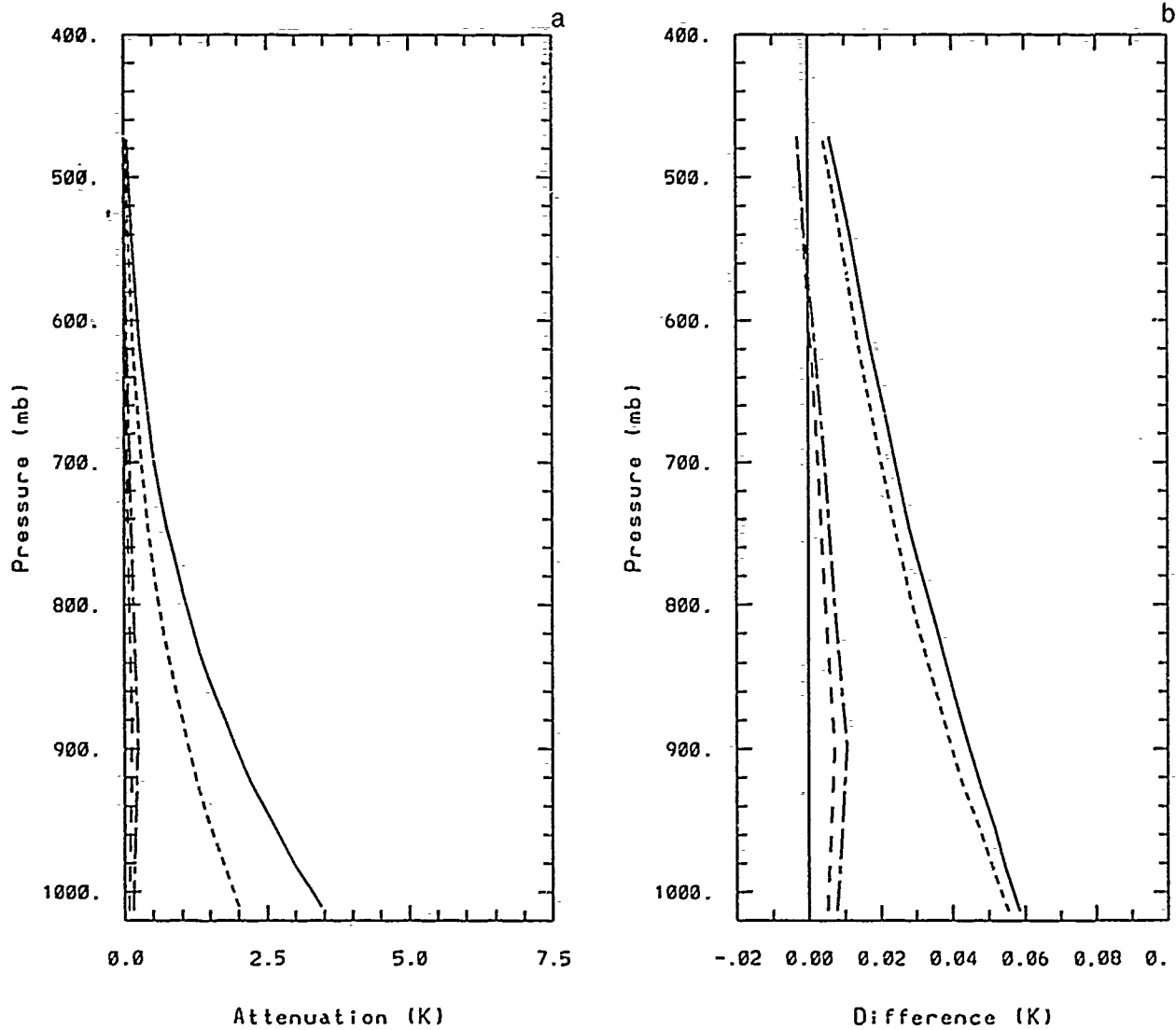


Figure 6.5 Same as Figure 6.3.

the upper-air temperatures at 9 standard levels between 1000 and 100 mb (1000, 850, 700, 400, 300, 250, 200, 150, 100 mb). The relative humidity is generally available up to 300 mb only. The algorithm converts the relative humidities to specific humidities and extrapolates the results to the 100 mb level by assuming a climatological water vapor concentration of 2.5 ppm at 100 mb (see Jursa, 1985). The upper-air data are subsequently interpolated to the pressure levels used for the numerical integration of the radiative transfer equation. Temperature is assumed to vary linearly with the logarithm of pressure while a P^n dependence is used for the specific humidities. The vertical stratification used by Susskind and Searl (1978) was found to provide good accuracy. Only the first 47 levels, from 1050 to 100 mb, have been retained. An additional level is used to represent the surface. The surface pressure is estimated based on local terrain elevation, provided by RTNEPH, and the standard pressure level heights from the forecast/analysis model.

The surface skin temperatures and terrain elevations are specified in RTNEPH on an 8th-mesh grid. Viewing angles are provided at quarter-mesh resolution. However, since the upper-air data do not change over a whole-mesh box (upper-air data is not spatially interpolated) there is no need to repeat the radiative transfer calculations for each 8th-mesh box. Instead, the $\tau_{m,i}$'s (see equation (4)) for all levels between the top of the atmosphere and the level corresponding to the lowest terrain elevation, are evaluated only twice per whole-mesh box, i.e. for view angles of 0 and 60 degrees. The $\tau_{m,i}$'s corresponding to the local value of the view angle is obtained by interpolating the transmittances between 0 and 60 degrees. This interpolation is done only once per quarter-mesh box. Finally, for each 8th-mesh box, the total atmospheric transmittance corresponding to the current terrain elevation is found by interpolating between neighboring adjacent levels. It is assumed that the logarithm of the transmittance depends linearly on the water vapor amount. As shown in Figure 6.6, the error introduced by this assumption is insignificant.

All possible sources of error being considered (transmittance modelling, vertical stratification, angular and vertical interpolation of the transmittances, and the neglect of stratospheric water vapor), it is found that the overall numerical precision of the radiative transfer scheme is better than .15 K.

The timing figures obtained on a 3 MIPS MicroVAX 3600 computer indicate that it takes no more than 1min 10sec of CPU time to complete the radiative transfer calculations for one Neph-box (64 whole-mesh boxes). This represents about a 100% increase in computation time when compared to the original RTNEPH code. Note that the computation time is roughly proportional to the number of atmospheric layers times the number of spectral subintervals used to cover the OLS-T band. While it is desirable at an

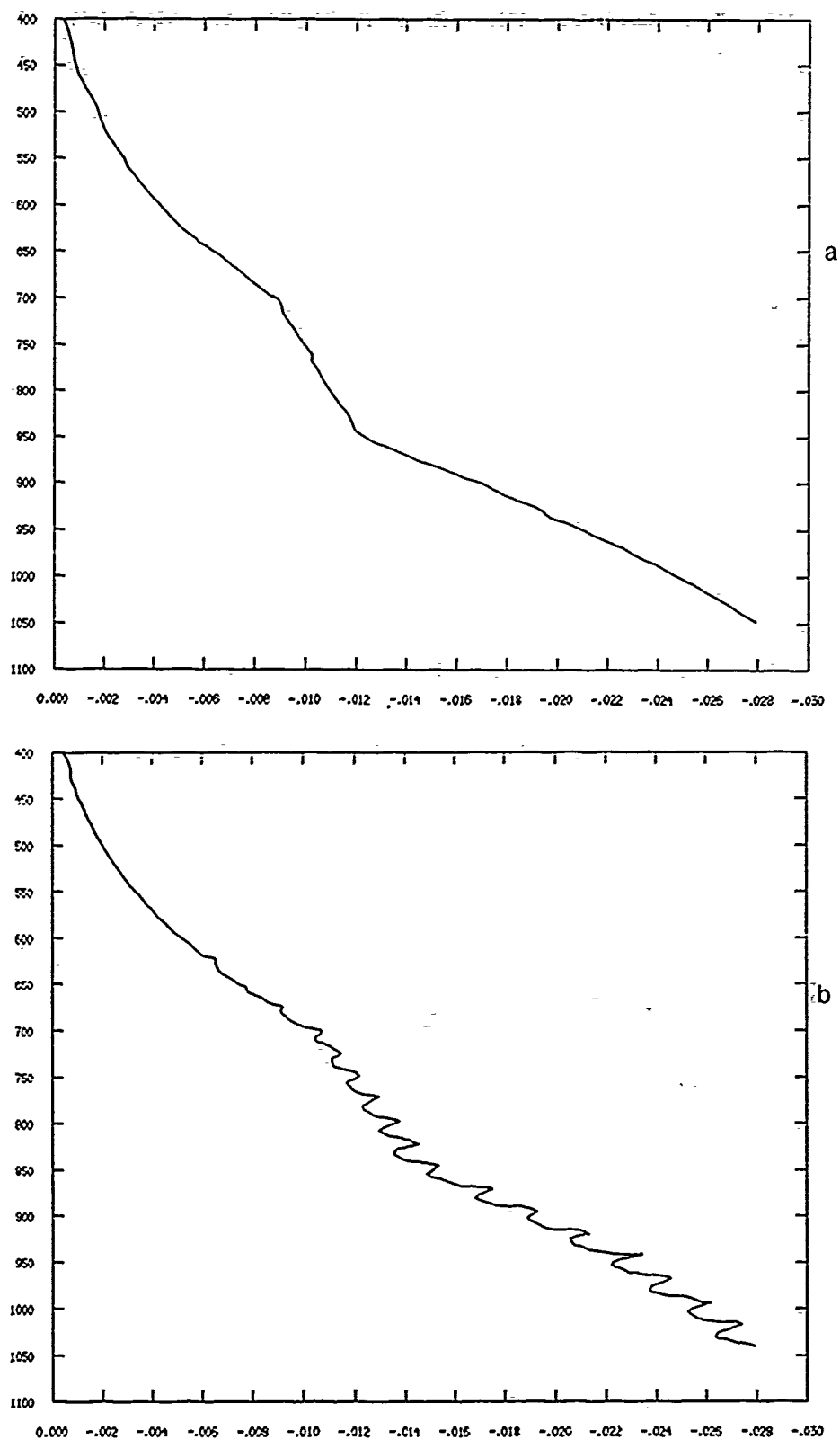


Figure 6.6 Differences between the profiles of atmospheric attenuation obtained with the RTNEPH version and the original version of the fast radiative transfer algorithm (see text), for a standard tropical atmosphere and for a scan angle of 45° . In 6a the surface temperature is equal to the ambient air temperature at all levels. In 6b, it is equal to the ambient air temperature +5 K.

experimental stage to maintain the best possible accuracy, provided that the computational load is not prohibitive, it is believed that one could reduce the vertical resolution as well as the spectral resolution of the radiative transfer model each by at least a factor 2 without severely impacting the accuracy of the results.

6.4 Evaluation of the New Water Vapor Attenuation Correction Procedure

6.4.1 The AFGWC Look-up Tables

The RTNEPH cloud/no-cloud decision is performed by comparing satellite measured brightness temperatures corrected for the atmospheric effect, to local estimates of the surface skin temperature. It is only when the difference between these two temperatures exceeds a certain threshold value that the presence of clouds is inferred. The threshold is normally introduced to account for uncertainties in the estimates of the clear-sky brightness temperatures; here, the threshold depends on satellite ID and grayshade. For measured temperatures above 260 K, the threshold value is 9 K. The RTNEPH brightness temperature correction, ΔT , is evaluated by summation of several correction factors,

$$\Delta T = \text{IR_Bias} + \text{Look_Angle_Bias} + \text{Solar_Zenith_Angle_Bias} \\ + \text{Tropical} + \text{Tuning}.$$

The magnitudes of these correction factors are determined from empirically derived look-up tables as a function of sensor ID and grayshade, geographical location, time and viewing geometry (see Table 6-3).

Table 6-3 Summary of look-up table dependence on physical parameters

	IR_Bias	Look_Bias	Zen_Bias	Tropical	Tuning	Threshold
Sat_ID	X	(X)	N	X	X	X
Grayshade	X		O	X		X
Scan angle		X	T			
Hemisphere		(X)				
Latitude			U	X		
Box#			S		X	
Water/Land			E		X	
YYDDD			D	X		
Day/Night					X	

The purpose of IR_Bias is to remove the systematic differences between the surface skin temperature and clear-column measurements. This correction factor includes atmospheric effects and presumably surface model biases and satellite calibration errors. It is supplemented in the tropics by an additional "Tropical" correction to account for the increased water vapor amounts in these regions. The latitude below which the "Tropical" correction is applied varies with the time of the year (YYDDD). An additional factor, Look_Angle_Bias, has been introduced to account for the increase of the water vapor path length at large viewing angles. Finally, a tuning factor is maintained for each Neph box to account for any remaining noticeable bias in the cloud analysis. Surface type and time of the day dependence is introduced into this factor in a crude way to compensate for potential deficiencies in the modelling of the surface skin temperatures.

Note that there is no separate correction table for the satellite calibration errors. Instead, these corrections are imbedded in the "atmospheric" correction and tuning tables, which explains why most factors depend on satellite ID and grayshade (see Table 6.3).

There are inconsistencies among the corrections applied in the five separate tables listed in Table 6.3. In particular, some parameters which relate to the same physical quantity are maintained independently in separate tables. For example, column water vapor amount is adjusted both by the IR_Bias and the Tropical corrections; however, the Look_Angle_Bias table, which is designed to account for scan angle effects, is completely independent of the water vapor amount in the other tables. Errors in the derived atmospheric correction are assumed to be somehow compensated for by the Tuning factors. Unfortunately, the Tuning tables are maintained on an irregular time basis and may be updated only when users focus their attention on a particular region where the corrections fail and the resulting cloud analysis is poor. It is also common to find inconsistencies in the magnitude of the tuning factors across adjacent grid boxes that have similar backgrounds. Because of these inconsistencies, and because it was not possible to tune the new GL model on the limited data available, it was decided not to use these factors in the subsequent analysis.

6.4.2 Comparison Between the Current AFGWC Model and the New GL Model

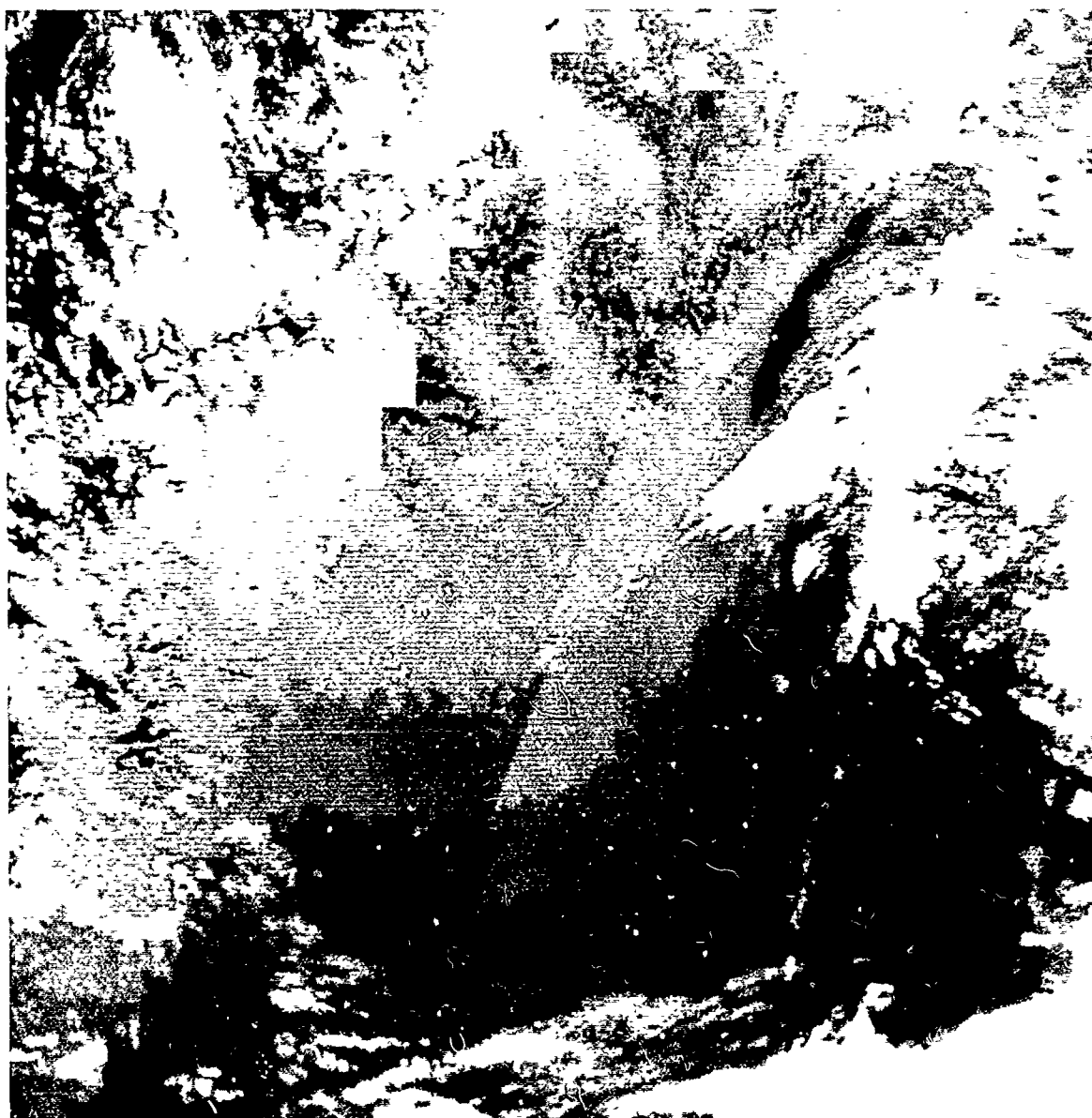
In this section we compare the current AFGWC model with the GL model which incorporates the new atmospheric correction procedure, based on OLS-T data obtained on 5 May 1990 between 0600 and 1400 UTC. The upper-air data needed for the radiative transfer calculations were taken from the 1200 UTC NMC global analysis (the data are not

interpolated to the local time of the satellite passage). The NMC model is a standard in the scientific community and has been widely documented (c.f. Rossow et al., 1989). According to Rossow et al., the accuracy of the temperature field is 3-4 K and the uncertainties in the water vapor amounts should be taken to be as large as 30%.

A cloud analysis was first performed using both models over 27 Neph boxes covering most of the Northern Hemisphere. The AFGWC run was made with the IR tuning factors removed from the look-up tables. Also, it was decided to run both the AFGWC and the GL models with the threshold turned off. The reason for this is that the correct magnitude of the threshold for the new model is undetermined. Also, the analysis is always biased towards finding less clouds as a result of thresholding. In such a case, whichever code produces the smallest estimate of the atmospheric attenuation tends to perform better in the sense that it detects more low clouds. This is regardless of how accurate the estimate of the atmospheric correction might be.

Figure 6.7 shows an example of a cloud analysis for Neph box 43 over the west coast of the United States. The corresponding attenuation fields produced by both models are included. This figure illustrates the artefact in water vapor attenuation corrections produced by the Tropical correction. The magnitude of the AFGWC correction increases wherever the Tropical correction is applied, with the result that the model misses the extensive field of marine stratus off the coast of the United States, south of the 28-degree latitude circle. Over that area, the threshold and tuning factor tend to compensate each other so that the effect is still apparent when both factors are turned on. Such an artefact is absent from the GL analysis. Note from the AFGWC attenuation field that the relative increase in the attenuation correction with scan angle is the same below and above the 28-degree latitude circle. The relationship between the Tropical correction and the Look-Angle correction is totally unphysical. Also apparent in Figure 6.7 is the change of grayshade with increasing values of the scan angle. It was discovered later in the comparison study that the raw OLS-T SGDB data include a one or two grayshade (1.9-3.8 K) correction according to the scan angle and the actual grayshade value. This correction is hardwired to the SGDB data acquisition process and is very poorly documented. One can expect that it is accounted for in the AFGWC model since the look-up tables have been derived based on the corrected data. However, it certainly affects the performance of the new scheme which, in its current form, expects raw measurements from calibrated sensors.

The cloud analysis for Neph box 52, which covers the region of the Gulf of Mexico, is shown on Figure 6.8. Here again the effect of the Tropical correction on the AFGWC analysis is obvious, especially over Florida. The tendency is for the AFGWC analysis to produce less clouds south of the 28-degree latitude circle. One striking difference between



a

Figure 6.7 Comparison of the cloud analysis produced by the AFGWC and the GL models for Neph box #43. The OLS-T image (a) is a mosaic of three different satellite passes at approximately 6 UTC, 8 UTC, and 13 UTC on May 5, 1990. The AFGWC and GL cloud/no cloud decisions are shown in 6.7-b and 6.7-c respectively. Figures 6.7-d and 6.7-e show the fields of brightness temperature correction produced by the GL model and the AFGWC model. The difference between those two fields is plotted in 6.7-f. The satellite viewing angles are shown in 6.7-g. The times of satellite passages are indicated on this figure. As it is apparent in Figure 6.7-e, AFGWC tropical correction coincide with the 28°N latitude circle.

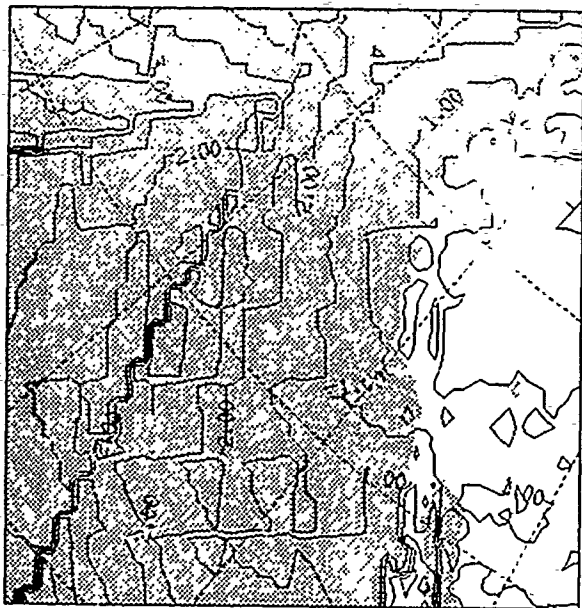


b

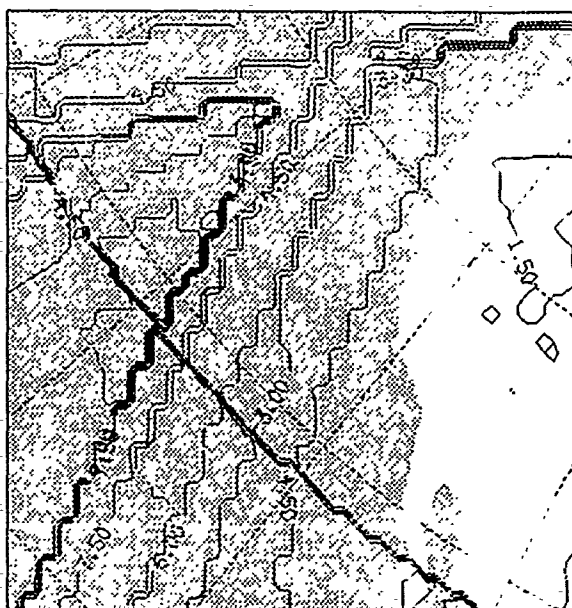


c

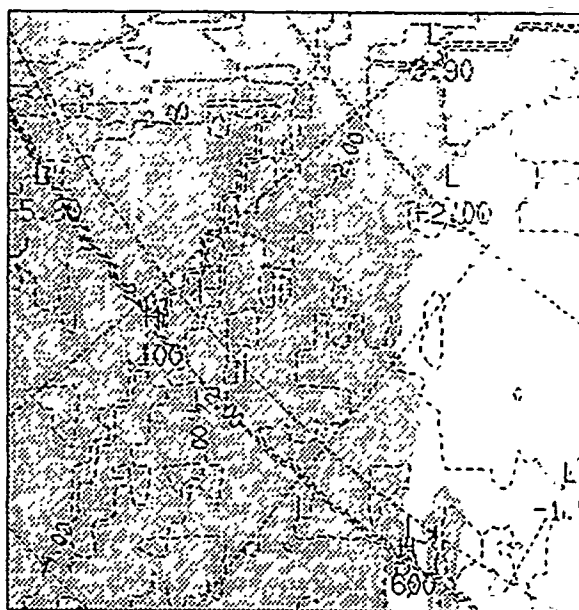
Figure 6.7 (continued).



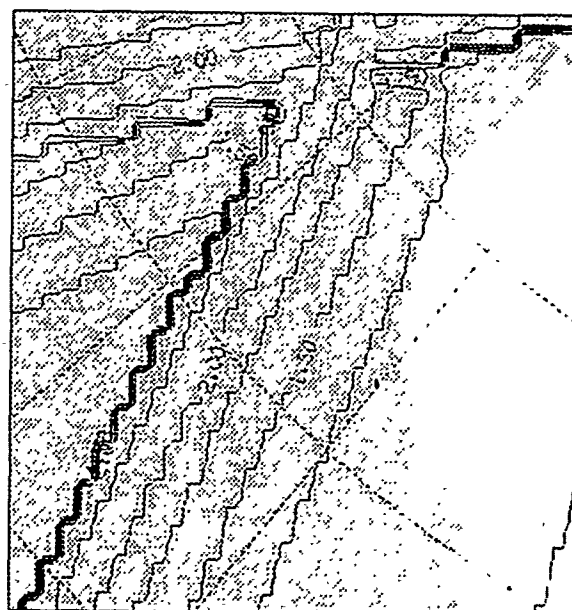
d



e



f



g

Figure 6.7 (continued).

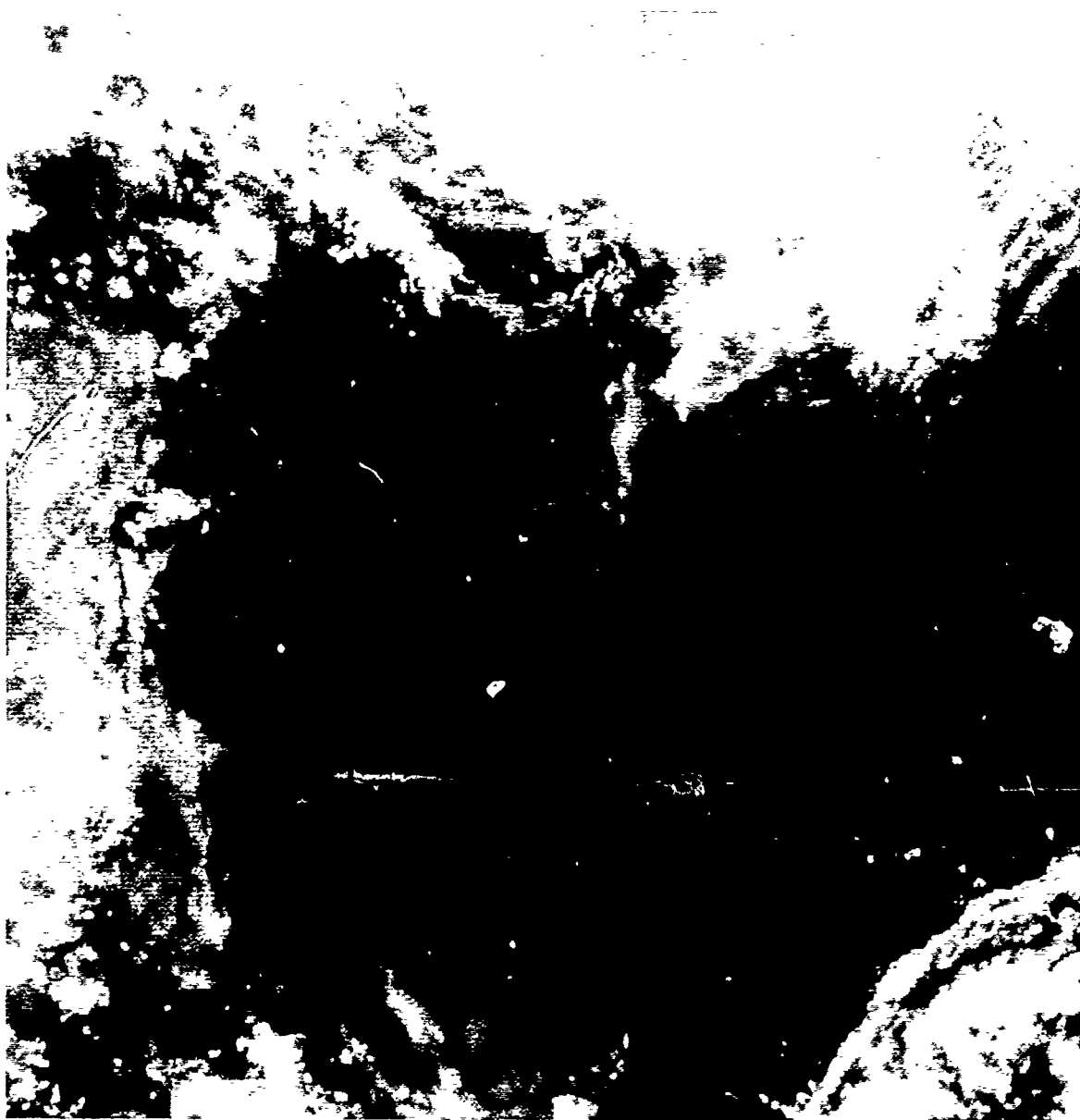
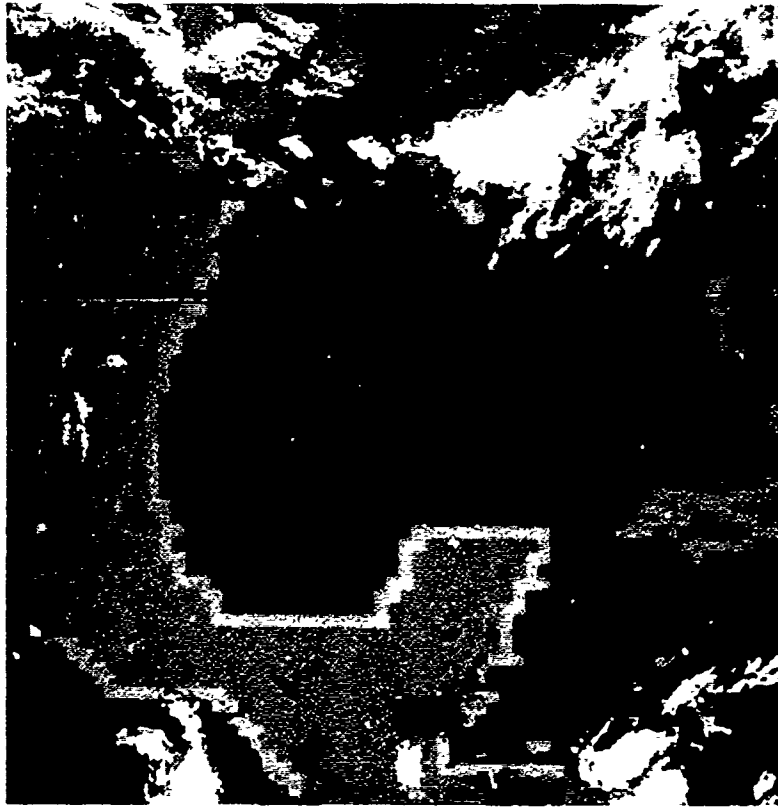
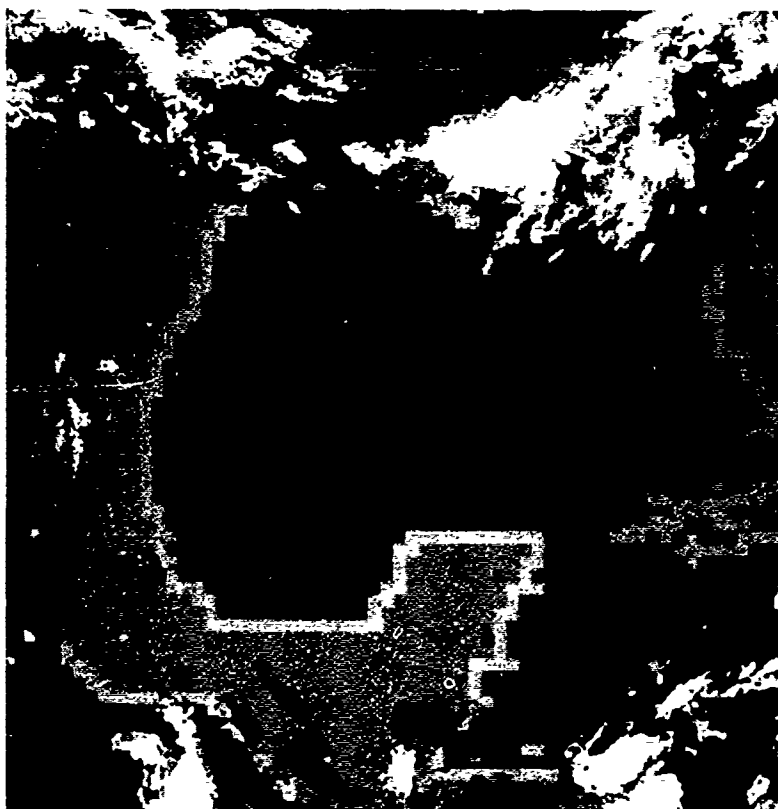


Figure 6.8 Same as Figure 6.7 for Neph box #52. As indicated in Figure 6.8g, the OLS-T data correspond to two satellite passes at approximately 12 UTC and 13 UTC, on May 5, 1990.



b

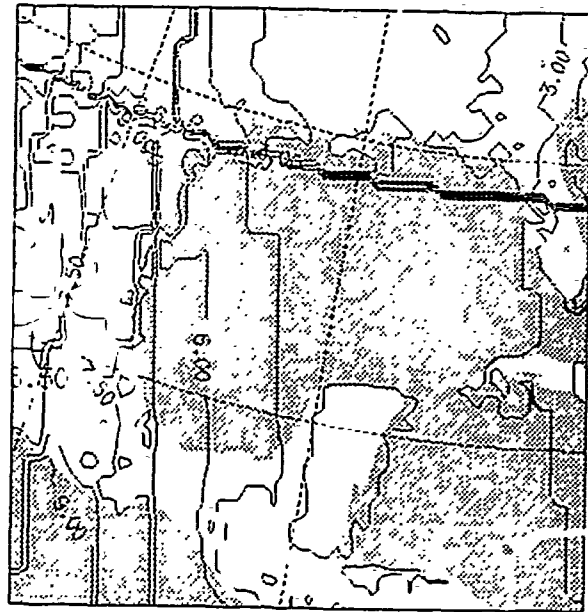


c

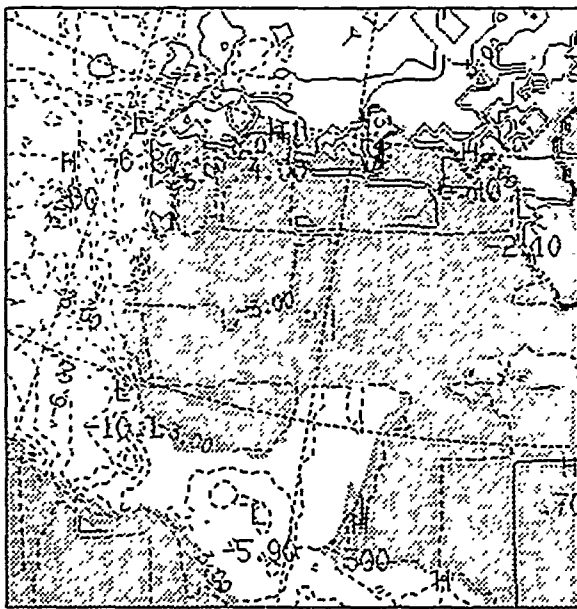
Figure 6.8 (continued).



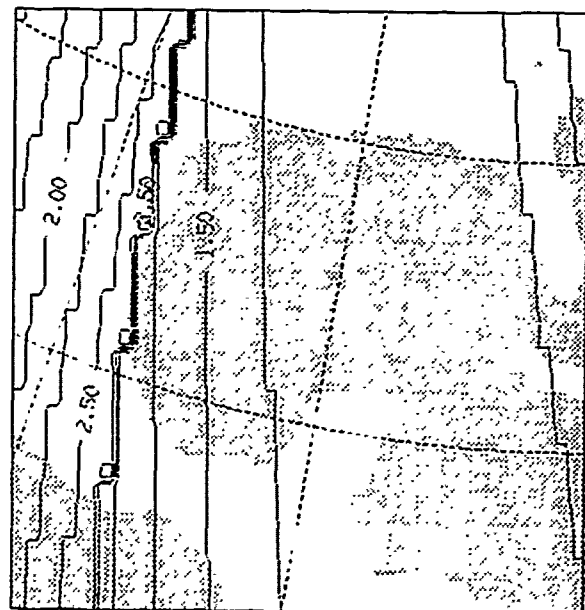
d



e



f



g

Figure 6.8 (continued).

the AFGWC and the GL models relates to the absence of land feature outlines in the AFGWC correction field. This results mostly from the fact that changes in terrain elevation are not explicitly taken into account in the look-up tables. The mistreatment of the terrain effect in the AFGWC model explains in particular the larger differences between the correction produced by the two models over the mountainous areas of central Mexico (the GL correction is smaller than the AFGWC correction in this area by as much as 6 K compared with 3 K over the nearby ocean). However, the impact on the AFGWC cloud analysis remains limited. The absence of low clouds in that area tends to mask the deficiency of the AFGWC model.

The few examples presented above are aimed at illustrating the inability of the AFGWC model to correctly reproduce the spatial variability of the water vapor attenuation effects and showing the extent to which the cloud analysis might be affected. The general observations made from an inspection of the complete set of 27 boxes are that more clouds are detected with the GL model especially in the tropics and in the areas of high terrain elevation. This is consistent with the result of a subjective analysis of the IR imagery. We noticed also that the number of clear areas affected by wrong cloud decisions even in the GL analysis remains marginally small in the absence of a threshold. This is an indication that some cold bias exists in the calculated clear brightness temperatures (which may originate from the surface skin temperature data base or possibly from the satellite data itself).

One must be careful not to conclude too rapidly about the performance of the models based on these results alone. For instance, the fact that the GL model detects more clouds in the tropical region is of little interest if the predicted clear brightness temperatures in both the GL and the AFGWC model only differ in the average by some constant value. In such a case, it should be possible to improve the quality of the analysis by simply reintroducing and adjusting locally the tuning factors for the current conditions in the AFGWC model or, as it is apparent later on in this section, by removing completely the brightness temperature correction. Of course, in the latter case it is not valid to conclude that application of no atmospheric corrections is superior to the modeled corrections; rather it is logical to search elsewhere for errors or biases (e.g., surface skin temperature data base). In fact, biases (systematic errors) in the modeled clear brightness temperatures should not be much of a concern as they can generally be corrected by careful tuning of the model. Of greater interest to the modeler is the magnitude of the standard deviation of the residual "random" errors, since it is precisely these random errors that the threshold is intended to account for. Reducing the variance of the errors would allow for a reduction in the magnitude of the threshold thereby enhancing the nephanalysis sensitivity to low or thin clouds that do not contrast well with the background. However, information on the variance of the random

errors cannot be obtained from a simple inspection of the cloud analysis. Rather it is necessary to analyze the difference between the brightness temperature produced by the model and the corresponding satellite sensor measurements.

The key goal in IR threshold decision techniques is to be able to predict as precisely as possible what the sensor would see at any particular time, over any particular area, in the absence of clouds. Therefore, the only sound and objective way to assess the performance of a particular algorithm is to directly compare the measured brightness temperatures to what the model predicts over some clear areas. Here again, one must rely on some sort of cloud decision (either manual or automated) in order to determine what area is clear. However, unlike the cloud truth, the chances of wrongful decision are lesser if one elects to reject those cases that are uncertain. Ideally, since systematic errors are dependent on both location and time, one should run separate time series analyses over different selected areas. Such analysis remains beyond the scope of the present study. Here, the objective is to provide sufficient evidence that the new algorithm is indeed performing better than the look-up table approach, based on a carefully selected subset of seven Neph boxes (Neph box # 30, 31, 38, 47, 52, 53, 54) for which manual cloud analyses have been produced.

Figures 6.9 to 6.15 show scatter plots of the differences between predicted and measured brightness temperatures as a function of viewing angle for clear areas over land (Neph boxes 30, 31, and 52) and over ocean (Neph boxes 38, 47, 53, and 54). The predicted temperatures are obtained from the AFGWC 8th-mesh surface temperature database, with the attenuation correction added. Each figure compares the errors in the calculated brightness temperatures obtained when: 1) no correction is applied; 2) with the AFGWC correction; and 3) with the GL correction. It is apparent from these figures that some cold bias is present in the modeled brightness temperatures. In some cases, the AFGWC tuning factor partially compensates for this bias (the values of the IR tuning factor are indicated on the figures). Recall that the scan angle correction mentioned earlier could not be removed. As a result, we do not observe a continuous drift of the error with increasing values of the scan angle as one would expect to see in the absence of atmospheric correction. Changes in the temperature bias due to the scan angle correction occur when the secant of the scan angle reaches values of 1.2 and 1.5, approximately. Other sudden changes in the temperature bias for land cases are generally attributed to the spatial correspondence between certain values of the scan angle and the occurrence of scenes with different surface properties. The ocean backgrounds are more homogeneous (both spatially and temporally) than land backgrounds, hence this effect is less dramatic over ocean boxes. We note also that the dispersion of the errors is greater over land than it is over the ocean. This may be explained by the fact that skin temperatures over land are empirically derived based on

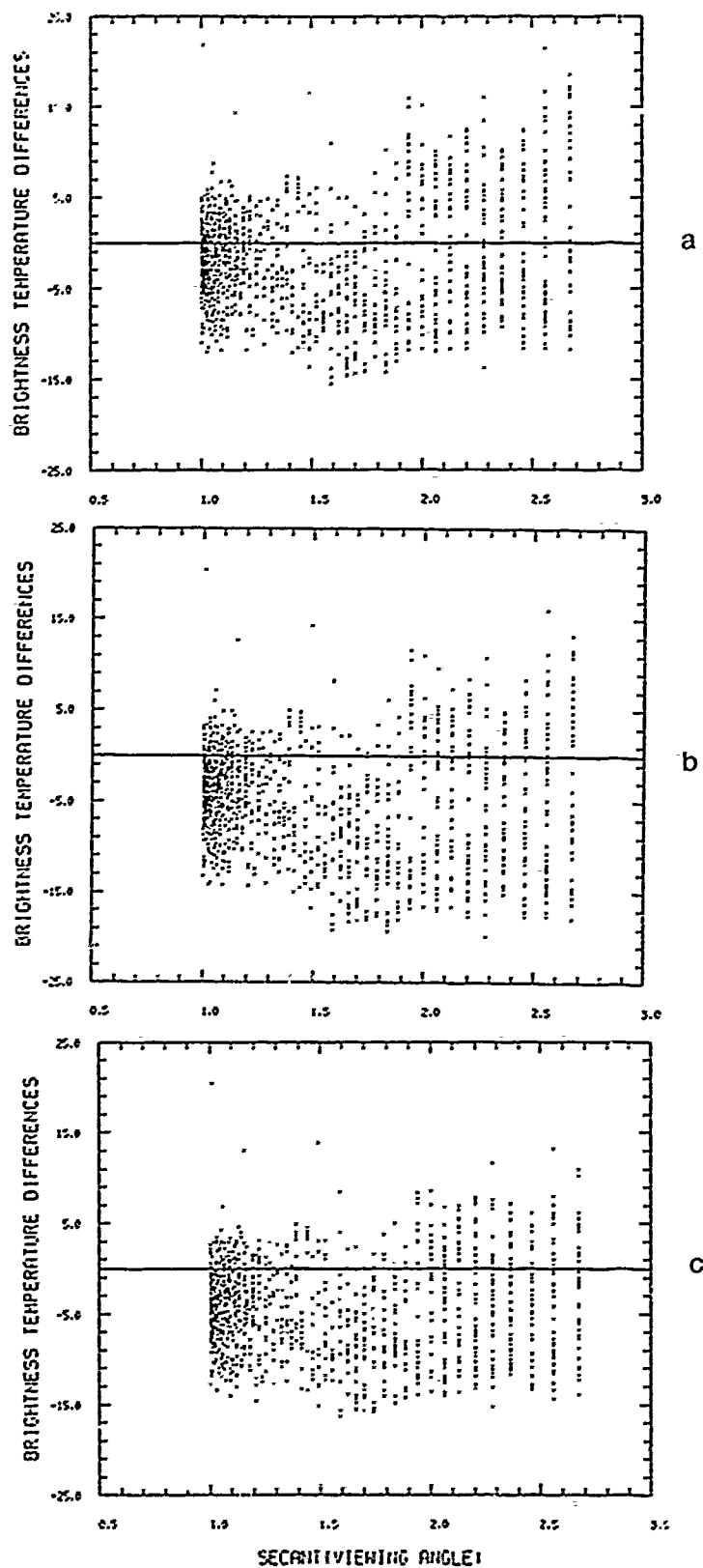


Figure 6.9 Scatter plots of the differences between modeled and measured OLS-T brightness temperatures over clear land areas in Eastern Europe (Neph box #30) versus the secant of the satellite viewing angle (i.e., relative path length). This figure shows the errors produced when the modeled skin surface temperatures are uncorrected, (a) and when the AFGWC correction (b) and the GL correction (c) are applied. In RTNEPH, a Tuning factor is normally applied to the satellite measurements to supplement the AFGWL correction. In this case, the value of the Tuning factor is -3.1 K. The data correspond to two satellite orbits at approximately 7 UTC and 9 UTC on May 5, 1990.

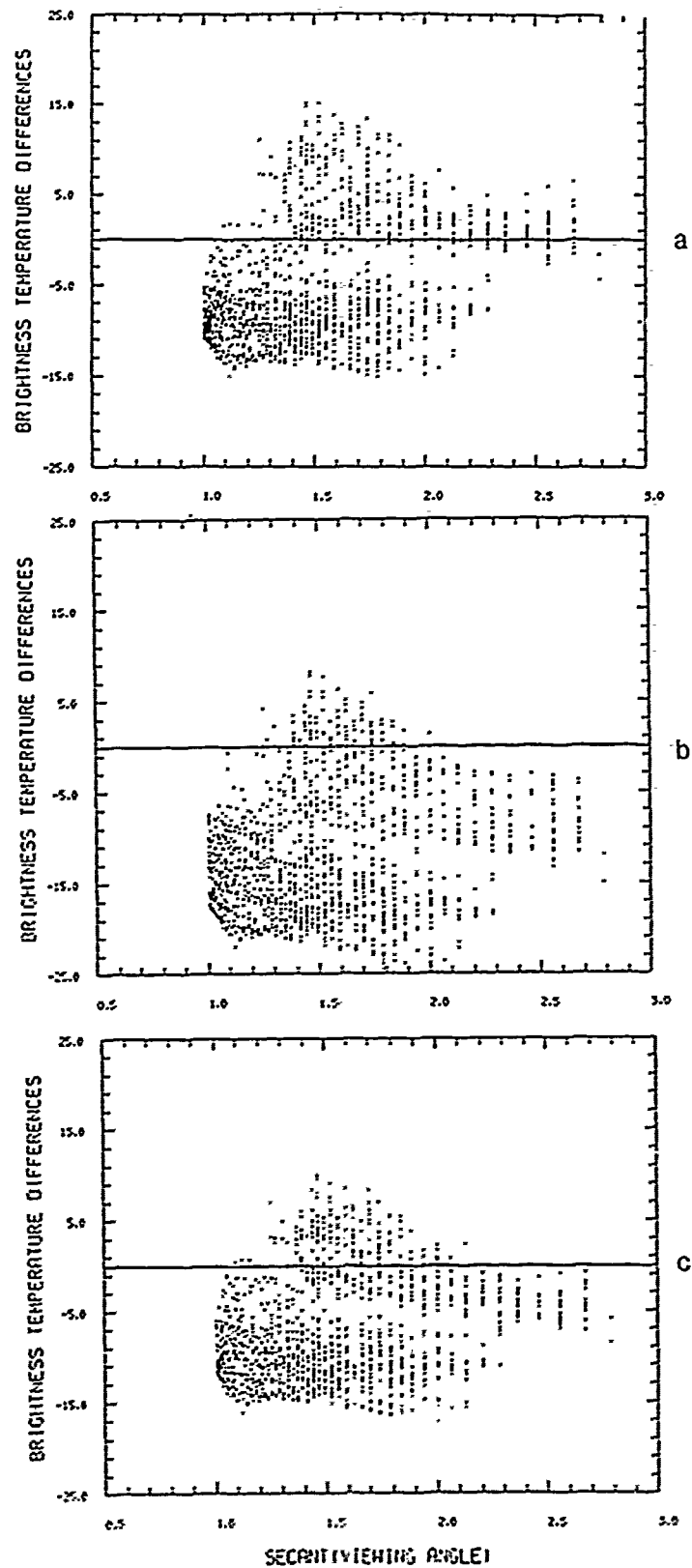


Figure 6.10 Same as Figure 6.9 but for the Eastern Sahara (Neph box #31). The data correspond to two satellite orbits at ~7 UTC and 9 UTC. In this case, the value of the Tuning factor is -3.1 K.

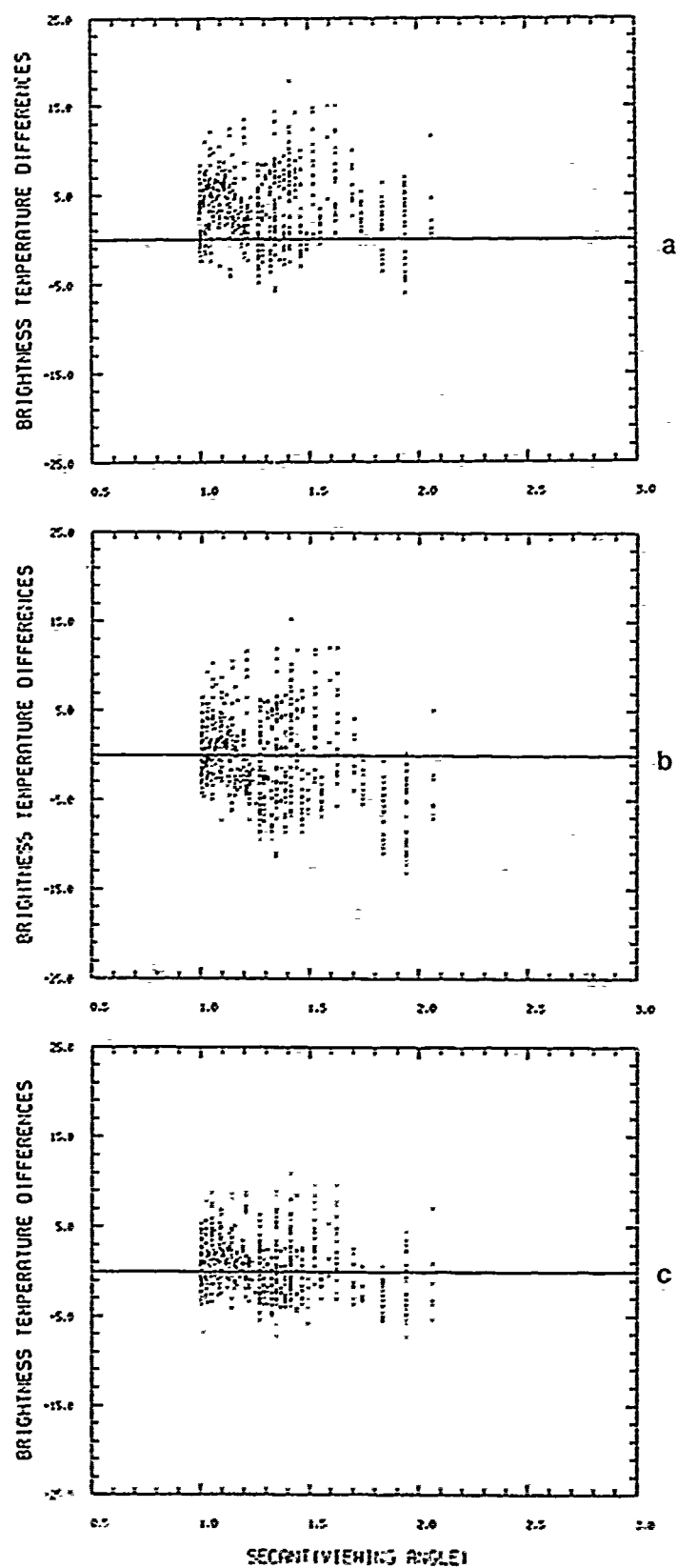


Figure 6.11 Same as Figure 6.9 but for Southeastern U.S. and Southern Mexico (Neph box #52). The data correspond to one satellite orbit at ~12 UTC. In this case the value of the Tuning factor is 0 K.

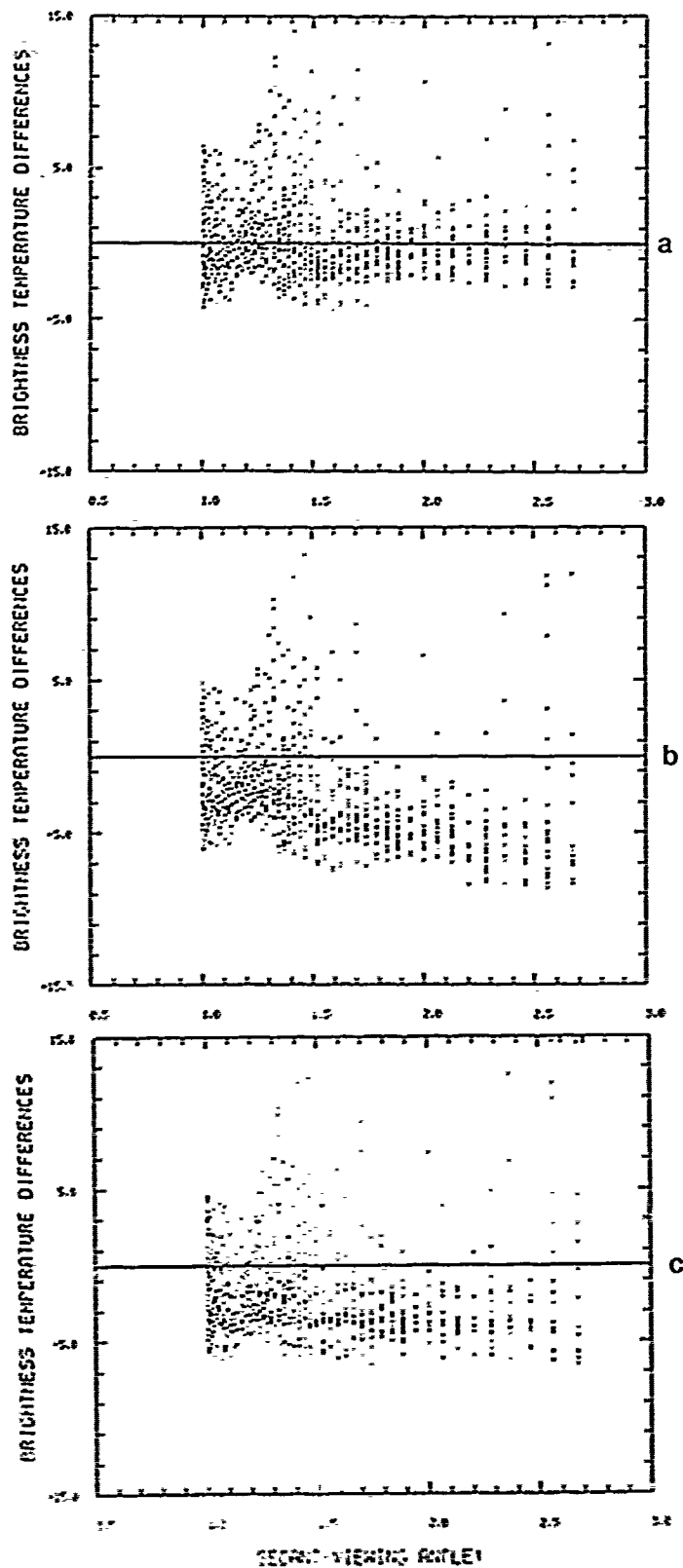


Figure 6.12 Same as Figure 6.9 but for North Atlantic ocean off the coast of Europe, west of 10°W (Neph box #38). The data correspond to one satellite orbit at ~ 11 UTC. In this case, the value of the Tuning factor is -3.1 K.

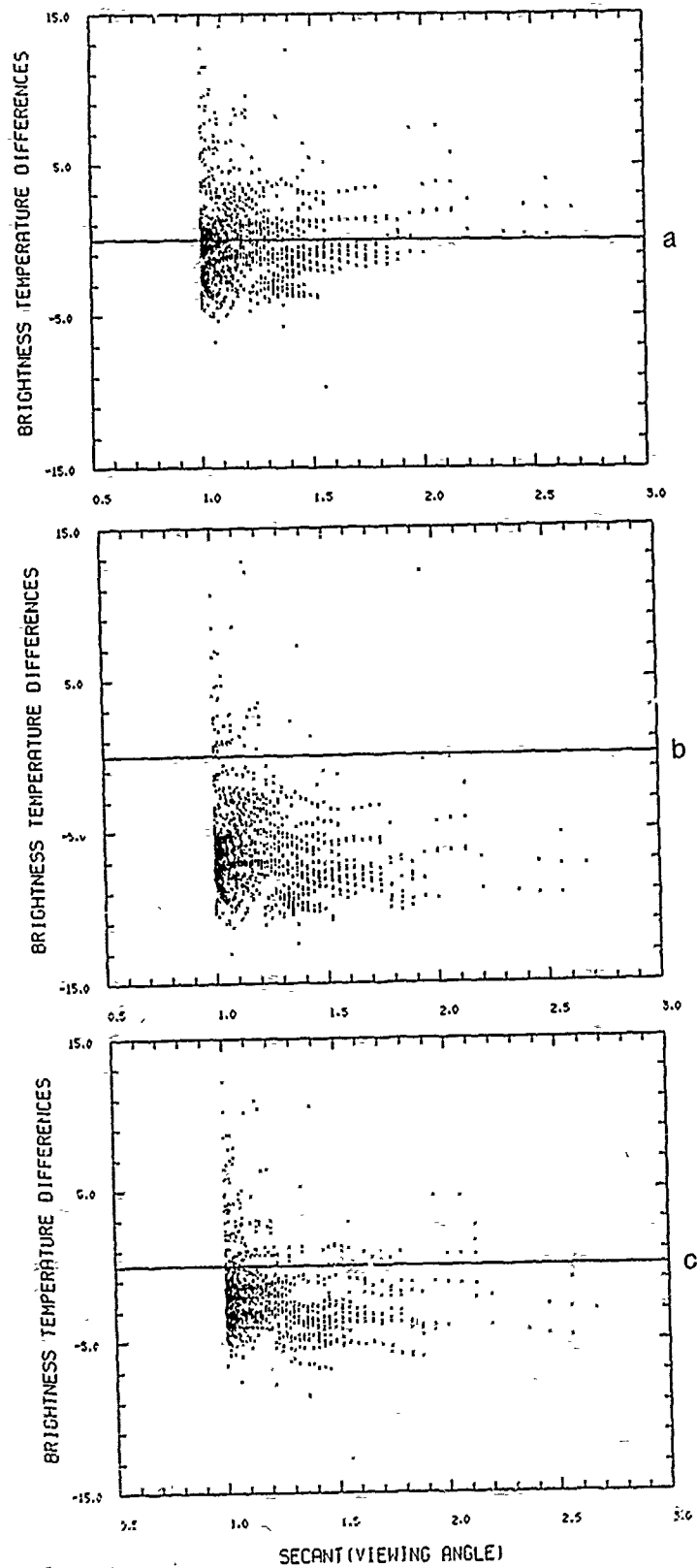


Figure 6.13 Same as Figure 6.9 but for Atlantic ocean, off the coast of North Africa (Neph box #47). The data correspond to one satellite orbit at ~11 UTC. In this case, the value of the Tuning factor is -7.6 K.

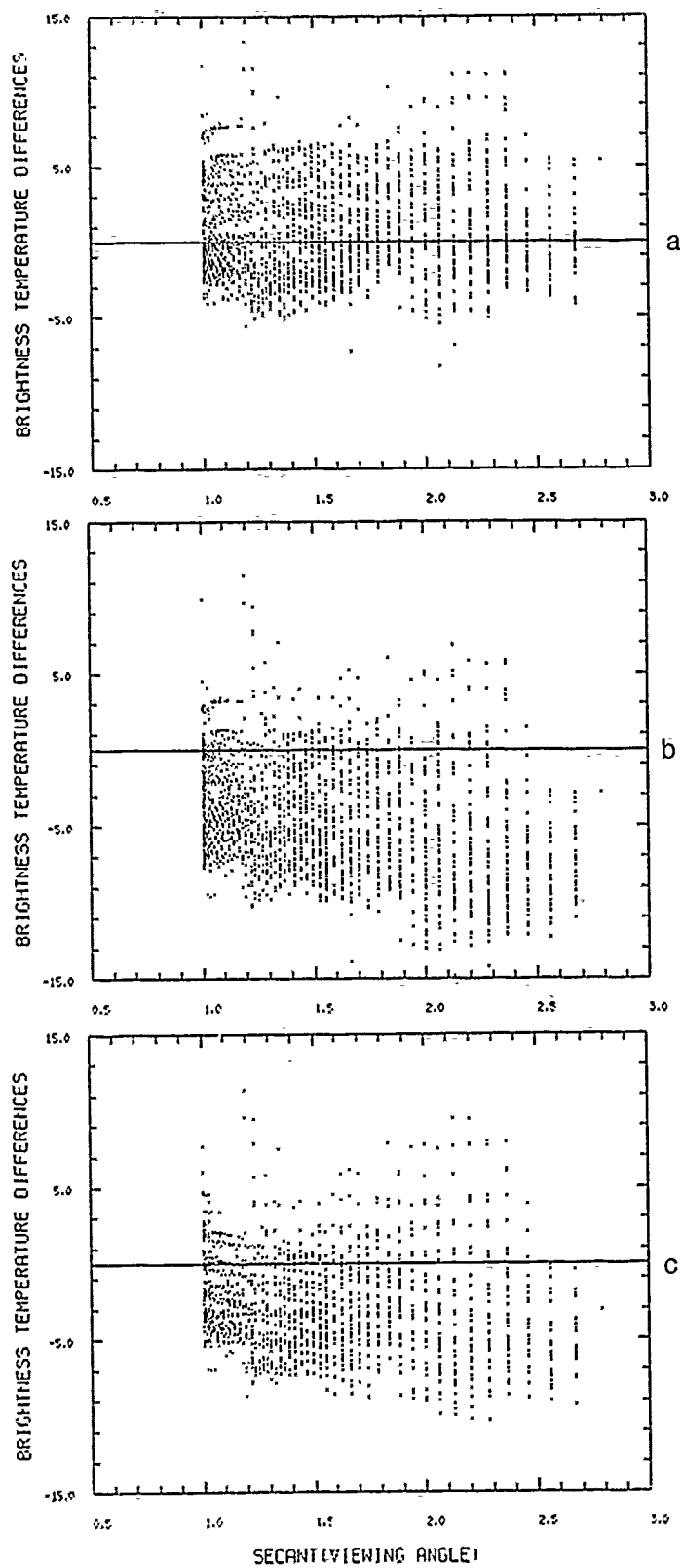


Figure 6.14 Same as Figure 6.9 but for Atlantic ocean and part of the Caribbean Sea (Neph box #53). Data correspond to three satellite orbits between ~10 UTC and 13 UTC. In this case, the value of the Tuning factor is -9.1 K.

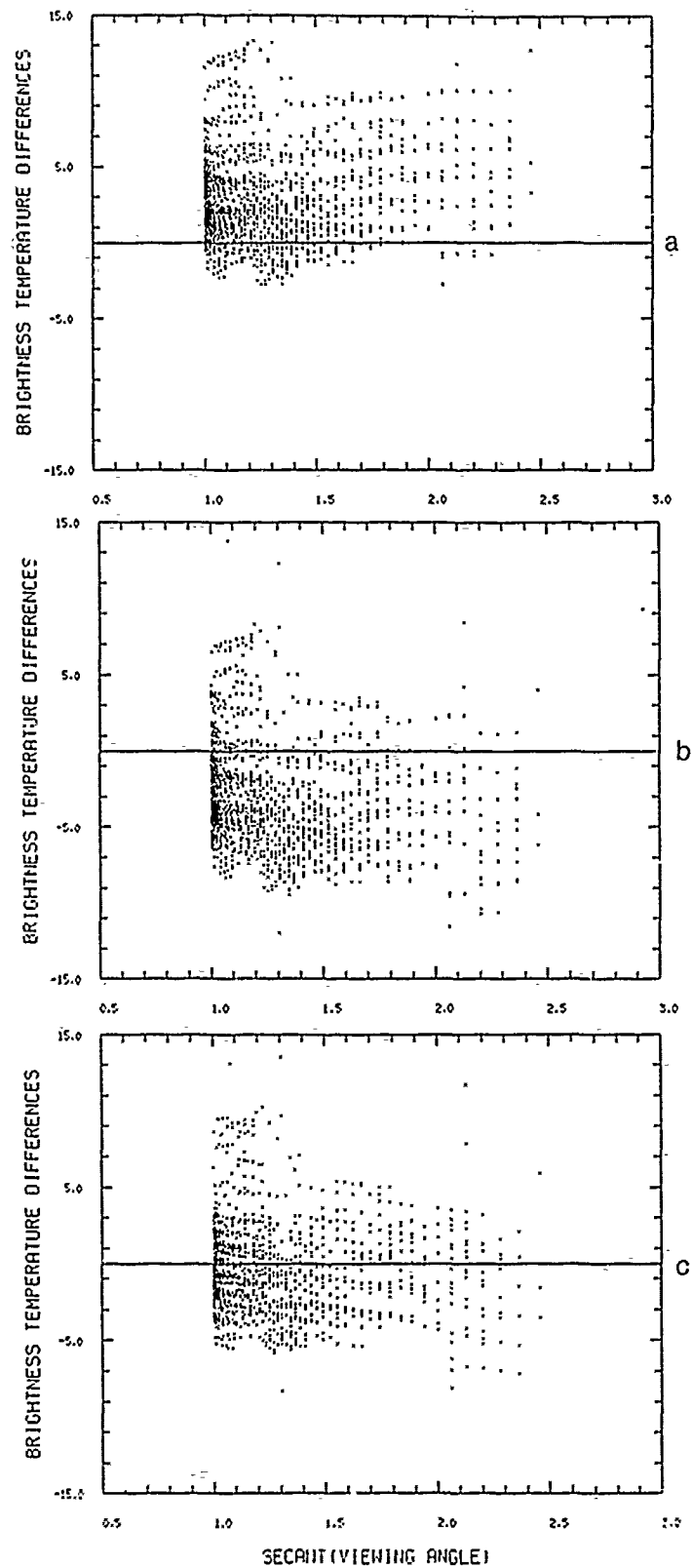


Figure 6.15 Same as Figure 6.9 but for Atlantic ocean (Neph box #54). Data correspond to one satellite orbit at ~12 UTC. In this case, the value of the Tuning factor is -4.6 K.

reported shelter temperatures whereas over the ocean they are obtained from more precise AVHRR clear-column multispectral measurements. Deficiencies in the modelling of the solar heating over land are also apparent in the scatter plots, especially in Neph boxes 30 and 31. In these figures we can observe two distinct groups of points. Each group corresponds to a different time of satellite observation with the colder group corresponding to an early morning passage of the satellite.

It is remarkable that the error patterns obtained with the GL model display in most cases the least dependence with respect to the scan angle. In the particular case of Neph box 52 already examined at the beginning of this section (see Figure 6.8), the sudden increase of the temperature bias in the AFGWC analysis for values of the secant of the viewing angle between 1.6 and 2.0 is an effect of the Tropical correction. All the points in this group, unlike the rest of the points, correspond to a region where the Tropical correction is applied. The most encouraging result is in the fact that the dispersion of the errors is considerably reduced when the new scheme is employed. This demonstrates the potential of the new GL model to provide smaller uncertainties in the predicted brightness temperatures after systematic errors are removed. The improvement is often more dramatic over land than over the ocean. One possible explanation is that the failure to reproduce the effect of changing terrain altitude over land in the AFGWC model contributes to a large fraction of the errors in the estimate of the atmospheric attenuation. Finally, it should be remembered that some further reduction in the error variance can be anticipated once a correction for satellite calibration errors is implemented in the GL model.

6.5 Summary and Conclusions

In the preceding sections, we described a new scheme for simulating DMSP/OLS-T brightness temperature measurements over clear areas that is intended for use in support of the RTNEPH cloud/no-cloud decisions. This scheme uses radiative transfer calculations to model atmospheric attenuation effects and is designed to accept standard upper-air data (temperature and humidity) from operational forecast or analysis models. The CO_2 absorption which, although it is small in the $11\mu\text{m}$ window, might not be negligible, as well as aerosol effects are not included in the model.

Decisions at various stages of the development of the radiative transfer model were driven by the desire to maintain the best possible numerical accuracy while reducing the computational requirements to a level judged acceptable for operational applications. The final design represents an increase of 100% in computation time when compared to the old "look-up table" version, which brings the processing time for a single Neph box from

1 minute to 2 minutes of CPU time on the AIMS computer. Unfortunately, as it was later discovered, the computational cost associated with the new water vapor attenuation scheme is substantially larger on the GWC computer (run time is increased from roughly 17 sec. to more than 2 min.) and it is such as to preclude an operational use of the new attenuation code on that machine. The fact that the performance of the GWC computer decreases significantly when executing floating point operations might possibly be at the origin of the timing problem (radiative transfer calculations, unlike the rest of RTNEPH, use floating point arithmetic). Since GWC has no plans for acquiring a new CPU in the near future and recoding the entire scheme would take a considerable amount of effort, alternate solutions have to be sought in order to be able to run the new version in real time. As mentioned earlier, computational efficiency of the new scheme can be improved (possibly by a factor 4) by degrading the spectral resolution and the vertical resolution of the radiative transfer model. The only effort involved is to recalculate the regression coefficients for the new spectral intervals. In the worst case, one might still consider using faster transmittance models like the one developed by McMillin et al. (1979).

A preliminary evaluation of the model has been performed based on a one-day set of satellite data in order to assess the impact of the errors in the transmittance calculations and in the supporting upper-air data. Although no quantitative evaluation could be made from such a small set of data, we presented some evidence that the new scheme might be used to improve the quality of the cloud analysis. These observations were based on the fact that the variance of the errors on the corrected satellite temperatures is demonstrably smaller with the new model. This is despite the fact that unlike the look-up tables, calibration errors are not taken into account. The implications are a potential for a reduction of the threshold over certain areas and therefore an enhanced detection capability of the low and thin clouds. It is worth noting that still better results would be expected if upper-air analyses were time-interpolated. It was found also that a cold bias exists in the GL analysis. As noted earlier in the report, these biases presumably originate from the surface temperature data base or from the satellite data itself (drift in the calibration curve or unknown grayshade correction).

Results obtained so far with the new water vapor correction scheme are encouraging. However, before the model becomes fully operational, a correction for the DMSP OLS-T sensor calibration errors should be implemented since these are not taken care of in the present model. Also the hardwired "scan angle correction" should be removed from the GWC ingest and/or mapping routines. It was argued that the latter correction might compensate for aerosol effects which are not treated in the model. If these effects are to be included, it would be better to try to model them rather than to apply an

arbitrary correction. The surface model should be investigated more thoroughly and an attempt be made to correct the biases either independently or in conjunction with sensor calibration errors (the two components cannot be separated unless some independent measurements are available). Finally, the cloud/no-cloud thresholds need to be adjusted. No real improvement in the cloud analysis can be expected from a more accurate estimation of the clear satellite brightness temperatures unless the thresholds are adjusted accordingly. The best approach to bias correction and threshold estimation is to directly compare modeled brightness temperatures to OLS-T data as was done in the present study. However, it is recommended that the analysis be performed on time series of data corresponding to several selected locations since the systematic errors in the atmospheric brightness temperature model are likely to depend on such factors as latitude, season, time of the day, surface type, etc. The corrections derived in this way would replace the current "Tuning" factors, the difference being that the former are obtained from objective analysis and, therefore, need not be readjusted periodically. Also, if the OLS sensors were calibrated it would be possible to use analyses from the multichannel AVHRR sensor to estimate the bias in modeled OLS brightness temperatures. One can take advantage of the improved cloud detection capability of the multichannel AVHRR instrument in the process of selecting those areas that are clear. Such an algorithm is currently being developed at GL. Model tuning would be done subsequently based on channel 4 or 5 data (or a combination of the two channels). However, in the absence of OLS-T calibration, this approach requires "co-located" OLS/AVHRR measurements in order to transform the tuning tables obtained from AVHRR IR measurements for use with the OLS-T sensor.

7. AIMS/AMPS Development

A large percentage of AER's responsibilities under the contract were concerned either directly or indirectly with software and hardware support for the AIMS/AMPS system. A considerable amount of software development occurred in support of both the general user community and specific program requirements. While many applications are supported on AIMS, the most significant in terms of resource allocation (i.e., manpower, computer requirements, image display and graphics) is nephanalysis.

AER responsibilities under the contract were directed primarily toward AIMS and consisted of system management support, hardware maintenance, system software development, specification and installation of new hardware and software, and system expertise for research applications. Initially AIMS system management was shared between AER and GL, however, as a result of personnel changes within GL, AER assumed sole responsibility for system management for the latter part of the contract.

7.1 System Description

AIMS/AMPS is an integrated system of distributed general purpose VAX and Encore processors, Adage imaging computers, and VAX-based monochrome and color workstations developed specifically to support research in remote sensing, principally satellite meteorology. Figure 7-1 is a schematic representation of the current system configuration. The most critical design feature of a distributed system such as this is the network architecture that links the various processors into a cohesive system. The AIMS/AMPS configuration provides an excellent solution to the networking problem through the use of a local area VAX cluster (LAVC) as the network backbone. LAVC is a DEC protocol implemented over ethernet that supports transparent access to all mass storage and processing resources on the network. In addition to the LAVC, several other networks connect the system to outside resources. A dedicated telephone line to the GL Ground Based Remote Sensing facility provides near real-time access to Doppler radar data. AIMS/AMPS is connected via the laboratory-wide broadband network to the GL main computing facility which in turn provides access to the Defense Data Network (DDN) and the NASA Space Physics Analysis Network (SPAN).

There are three primary data sources for AIMS. Most nephanalysis-related data are obtained via magnetic tape from either AFGWC, ETAC, or NESDIS. Real-time direct readout satellite data are routinely acquired at the AIMS GOES ground station; there is also a limited NOAA Automatic Picture Transmission (APT) capability. Global meteorological

observations in the form of surface, upper air, manually digitized radar reports, and NMC model guidance are received and decoded automatically via a WB604 data line. Satellite data dominate all other data types in competition for system resources. A primary system design consideration was management of the vast amount of imager and sounder data available from the satellite sensors.

Principal access to the system for interactive applications is through the VAX workstations and Adage imaging computers. Software has been developed to perform standard display operations such as grayshade imaging, pseudocolor enhancements, time-series looping, and graphical display. Also, the Adage systems support full color display of multispectral imagery as well as fast interactive image enhancement and filtering operations. In addition to these general capabilities, a large body of applications software has been developed to support specific research requirements for visualization of analysis results and quality control of input data. Kleespies et al. (1987) summarized the various research applications that were being supported by AIMS in 1987. Since then a number of new projects have been added including: a validation study of SSM/I retrieval algorithms (Hollinger, 1989; Gustafson and Felde, 1989), an investigation of AVHRR and GOES split-window techniques for retrieval of low level water vapor and precipitable water amounts (Kleespies and McMillin, 1990), and comparisons of a new cloud layering algorithm relative to the RTNEPH clustering algorithm (d'Entremont et al., 1989).

7.2 System Expansion

During the course of the contract, the AIMS/AMPS system grew significantly. AER responsibilities relative to system growth were varied. They included identification and specification of expansion requirements, identification of specific hardware and software products that were available to satisfy the requirements, arranging for on-site demonstrations and trial loans, presentations to users of various upgrade options, monitoring, and in some cases performing, installation of new products, and reconfiguration of the system to accommodate new products. Hardware additions that AER was directly involved with were: 3 VAXstation 2000 workstations, 2 VAXstation 3600 workstations, a Tektronics vector rasterizer, an 8 mm cartridge tape drive, 3 ethernet terminal servers, 5 VAXstation 3100 workstations, a VAXserver 3800 as boot node, an X-windows terminal, 2 magneto-optical disk drives, a line printer, 2 laser printers including 1 postscript printer, and an ink jet color printer. Mass storage capacity grew from approximately 1.5 Gbytes to approximately 6.28 Gbytes system wide, exclusive of the optical disks. As an alternative to adding more nodes to the LAVC, one of the microVAX II computers was recently upgraded

to a 3800 class machine through a CPU replacement. A second machine is scheduled for upgrade in the next year.

Software additions and upgrades also occurred to keep pace with new revisions and with changes in hardware. A major software upgrade to version 5.x of the VMS operating system was performed on all VAX systems midway through the contract. Initially version 5.1 was installed; a less disruptive upgrade to version 5.3 was performed later. The MPX operating system on the Encore ground station computer was also upgraded to version 3.3. C language compilers are now licensed throughout the cluster in addition to FORTRAN. GKS has been settled on as the de facto graphics standard on the system, at least for the present. All layered products (e.g., compilers, editors, VMS utilities, etc.) have been upgraded at least once for maintenance and product enhancement revisions, some several times. A coverage and performance analysis software package was added to the system to assist in software development. Included in version 5.x of VMS was the Digital implementation of the emerging X-windows standard known as DECwindows. While offering great potential for windowing applications, acceptance among AIMS/AMPS users was slow due to apparent performance problems with the initial implementation. However, addition of more memory to the workstation computers along with improved understanding of software features has since led to the implementation of DECwindows on most workstations in the cluster.

The process of integrating all of the hardware and software expansions into the AIMS/AMPS system has required careful preparation and expansion of support capabilities. Virtually the entire Satellite Meteorology Group and several members of the Atmospheric Prediction Group have their offices connected to the system via thin-wire ethernet either directly or through terminal servers. Thin-wire connections were established for each of the workstations as they came online. System management responsibilities have grown both with the number of nodes on the cluster and with the number of users. With the addition of each workstation or computer node it is necessary to reconfigure and tune the entire AIMS system. Recent emphasis has been on automating as many system management functions as possible to reduce the impact of normal system growth down to a level that can continue to be maintained by two people.

7.3 System Capabilities

A number of capabilities were added to the system. The Adage imaging computers were physically connected to dedicated VAX workstation computers from their original microVAX configurations. This eliminated previous bottlenecks on AIMS when two users

were using the Adage systems simultaneously. It also added the graphics and windowing capabilities of the workstations to the Adage image processing capabilities. A black-and-white image hardcopy capability was implemented using a surplus laser FAX recorder. A reliable image recording capability from an Adage monitor onto VHS video tape was developed. This subsystem uses an NTSC encoder for conversion of the RGB signals and an external frame synchronizer to prepare the Adage image output for recording on a video cassette recorder.

User terminals were removed from the unreliable laboratory-wide broadband network and put on the AIMS/AMPS ethernet via terminal servers. Eventually the AIMS/AMPS ethernet was connected via a bridge to the main GL computing center and other laboratory facilities by a separate ethernet installed on the broadband. That brought direct access to both SPAN and DDN to AIMS/AMPS users along with fast and reliable DECnet access to the computing center VAX systems. The latter is particularly significant following the recent GL acquisitions of the high speed VAX 9000 main-frame and VAX 6000 mini computers. A separate, direct link to the GL ground based remote sensing radar site was also established via a dedicated 9600 baud serial line.

An APT ground station was developed using existing hardware to receive direct broadcast NOAA AVHRR and Soviet METEOR image data transmissions. The ground station uses a hardware interface shared with the laser FAX image hardcopy recorder described above. Device driver and real-time data acquisition software were written to support two-channel AVHRR and one channel METEOR ingest.

Recently the State University of New York (SUNY) lightning detection network was connected to AMPS through ethernet via a terminal server connection to a SUNY network end node maintained outside of the AIMS cluster by IYA. AER assisted with specification and development of ingest and server software.

7.4 Summary

Growth of the combined AIMS/AMPS computer system has been explosive over the term of the contract. AER has been closely involved in maintaining the system through system management responsibilities and hardware technical support, and in system expansion through recommendations and investigations of new products, integration of new hardware and software, and monitoring of system performance. System capabilities have been continuously expanded in response to research requirements. Unlike many centralized computer systems, AIMS/AMPS was developed, and has been maintained, with the single goal of supporting the research users. As such it has evolved into a highly

specialized system suited to interactive processing, analysis and display of satellite and radar data and to the associated support functions required by its user community.

8. References

- Anderson, G.P., S.A. Clough, F.X. Kneizys, J.H. Chetwynd, and E.P. Shettle, 1986: AFGL Atmospheric Constituent Profiles (0-120 km). Air Force Geophysics Laboratory Technical Report AFGL-TR-86-0110, Hanscom AFB, MA 01731-5000, 43 pp. ADA175173.
- Bianco, A. and H.-C. Huang, 1990: Satellite or radar forecast techniques for short term prediction of storm motion on the Remote Atmospheric Processing and Display (RAPID) System. GL-TR-90-0179.
- Clough, S.A., F.X. Kneizys, L.S. Rothman, and W.O. Gallery, 1981: Atmospheric Spectral Transmittance and Radiance: FASCODE 1B. Proc. SPIE, 277, Atmospheric Transmission.
- Clough, S.A., F.X. Kneizys, and R.W. Davies, 1989: Line Shape and the Water Vapor Continuum. Atmospheric Research, 23, 229-241.
- Cotton, W. R., R. McAnelly, C. Tremback, and R. Walko, 1988: A Dynamics Model for Forecasting New Cloud Development. AFGL-TR-89-0011. ADA213939.
- d'Entremont, R.P., G.B. Gustafson, J.T. Bunting, M.K. Griffin, C. Barker-Schaaf, P.L. Novak, and J.M. Ward, 1989: Comparisons between the RTNEPH and AFGL Cloud Layer Analysis Algorithms. AFGL-TR-89-0175. ADA216637.
- Gustafson, G.B. and G.W. Felde, 1989: Validation of automated cloud detection from microwave imagery. Preprints, 5th Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology. AMS, Boston, MA.
- Heideman, K.F., H.-C. Huang, and F.H. Ruggiero, 1990: Evaluation of a cloud nowcasting technique based on GOES IR satellite imagery. Preprints, 5th Conference on Satellite Meteorology and Oceanography, AMS, Boston, MA.
- Hoke, J., J. Hayes, and L. Renninger, 1981: Map Projections and Grid Systems for Meteorological Applications. AFGWC Technical Note 79/003, Offutt AFB, Nebraska, 86 pp.
- Hollinger, J., 1989: DMSP Special Sensor Microwave Imager Calibration/Validation. Naval Research Laboratory Final Report, Volume I.
- Jursa, A.S., editor, 1985: Air Force Handbook of Geophysics and the Space Environment. #ADA167000.
- Kleespies, T.J. and L.M. McMillin, 1990: Retrieval of precipitable water from observations in the split window over varying surface temperatures. J. Appl. Meteor., 21, 9, 851-862.
- Kleespies, T., R. d'Entremont, G. Felde, K. Hardy, L. Thomason, G. Gustafson, and C. Ivaldi, 1987: Application of the AFGL Interactive Meteorological System to Atmospheric Research. Preprints, 3rd Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology, AMS, Boston, MA.

- Liou, Y.-C., 1989: Retrieval of the three-dimensional wind and temperature fields from one component wind data by using the four-dimensional data assimilation technique. Master's Thesis, The University of Oklahoma.
- McMillin, L.M., H.E. Fleming, and M.L. Hill, 1979: Atmospheric Transmittance of an Absorbing Gas. 3: A computationally fast and accurate transmittance model for absorbing gases with variable mixing ratios. Applied Optics, **18**, 1600-1606.
- Rossow, W.B., C.D. Garder, and A.A. Lacis, 1989: Global, Seasonal Cloud Variations from Satellite Radiance Measurements. Part I: Sensitivity Analysis. J. of Clim., **419-458**.
- Ruff, I. and A. Gruber: 1975: Graphical Relations Between a Satellite and a Point Viewed Perpendicular to the Satellite Velocity Vector (Side Scan), NOAA Technical Memorandum NESS 65, U.S. Department of Commerce, Washington, DC, 14 pp.
- Schaaf, C.L., J.M. Ward, G.B. Gustafson, R.P. d'Entremont, M.K. Griffin, P.L. Novak, and J.T. Bunting, 1990: Effects of terrain elevations on the RTNeph IR satellite processor. Preprints, 5th Conference on Mountain Meteorology, AMS, Boston, MA.
- Susskind, J., and J.E. Searl, 1978: Synthetic Atmospheric Transmittance Spectra near 15 and 4.3 μm . J. Quant. Spectrosc. Radiat. Transfer, **19**, 195-215.
- Thomason, L. and C. Ivaldi, 1990: A unified treatment of conventional meteorological data. Preprints, 6th Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology, AMS, Boston, MA.
- Tremback, C.J., G.J. Tripoli, and W.R. Cotton, 1985: A regional scale atmospheric numerical model including explicit moist physics and a hydrostatic time-split scheme. Preprint, 7th Conference on Numerical Weather Prediction, June 17-20, 1985, Montreal, Quebec, AMS.
- Weinreb, M.P., and A.C. Neuendorffer, 1973: Method to Apply Homogeneous-Path Transmittance Models to Inhomogeneous Atmospheres. JAS, **30**, 662-666.
- Weinreb, M.P., and M.L. Hill, 1980: Calculation of Atmospheric Radiances and Brightness Temperatures in Infrared Window Channels of Satellite Radiometers. NOAA Technical Report NESS 80, U.S. Dept. of Commerce, Washington, DC, 40pp.

Appendix A: The updated GL-3D job control file

\$INDAT

! Variables to change for BOTH models

TIMMAX = 7200., ! Ending time of model integration (seconds)
FRQPRT = 1800., ! Frequency of printing/plotting model fields(sec)

! Variables to change for FULL model

! These variables control the reading and writing of history files for
! the full model. The mixed model is fast enough where it can just
! be rerun. The full model can write a history file, then restart
! from the file at a time that was written on the file.

IOOUTPUT = 0, ! Flag for writing history file
! (0-no files, 1-write and save files)
FRQHIS = 3600., ! Frequency of writing to history file (sec)
IFILSTR = 0, ! (0-initial start, 1-history file restart)
TIMSTR = 7200., ! Time of history restart
HFILIN = 'SSA7HC', ! File specification of history restart file

MAXHFL = 10, ! Maximum number of time slices on each history
! file
NHWDMX = 10000000, ! Maximum number of words on each history file
! Next output file will be used if these are
! reached

HFILOUT(1)='SSA5HA', ! First history file spec (up to 10)
HFILOUT(2)='SSA5HB', ! Second history file spec
! !!! Output files cannot be the same name as
! !!! input files !!!!!!!

! Variables to change for BOTH models

NNXP = 21, ! The number of grid points in the x-direction
NNYP = 21, ! The number of grid points in the y-direction
NNZP = 10, ! The number of grid points in the z-direction

NNQPARM = 0, ! Flag for the convective parameter (1-on, 0-off)

! Grid and time step specifications

DTLONG = 10., ! Timestep length (sec)

NRATIO = 4, ! Ratio of long to small timestep (integer)
DELTAX = 5000., ! Grid spacing in x-direction (meters)

DELTAY = 5000., ! Grid spacing in y-direction (meters)
 DELTAZ = 400., ! Lowest grid spacing in vertical (meters)
 DZRAT = 1.5, ! Stretch factor of vertical grid spacing
 DZMAX = 750., ! Maximum vertical grid spacing

SURFILE='RAMS:RADTOP.DAT', ! Input surface characteristic file
 ! See SETUP.DOC for info

SNDFILE='RAMS:DVR1_1530.SND', ! Input sounding file
 ! See SETUP.DOC for info

NUDFILE='RAMS:C1_1415.DAT', ! Input Doppler wind file
 ! See SETUP.DOC for info

! Nudging controls

STRTNUD = 0., ! Model time to start nudging
 ENDNUD = 3600., ! Model time to end nudging
 FRQNUD = 10800., ! Optional control to change nudging data
 ! during a run. Currently not implemented.
 NUDRAD = 2, ! Nudging type (0-no nudging 1-radial nudging
 ! 2-Cartesian nudging)
 SCLNUD = 1800., ! Nudging timescale
 IDR = 7, ! x grid point location of radar for radial nudge
 JDR = 12, ! y grid point location of radar for radial nudge

STRTIM = 14.875, ! Local solar time at beginning of simulation
 RLAT = 39.3, ! Latitude of southern edge of domain
 WLON1 = 105.25, ! Longitude of western edge of domain
 ! (positive north and west)

! Cumulus parameterization parameters

CONFRQ= 1800., ! Frequency of convective tendency update
 WCLDBS = .02, ! Minimum resolved vertical motion at LCL for
 ! convection (m/s)

 ! Variables to change for FULL model

CPHAS = 20., ! Phase speed for lateral boundary conditions (m/s)

! Radiation parameters

NIRAD = 1, ! Flag for the radiation parameter (1-on, 0-off)
 ! Note : Must run radiation if soil model is run.
 ! Note 2: Radiation takes a long time if high
 ! resolution input sounding is used.

RADFRQ = 600., ! Frequency of radiation tendency update (sec)
 IMONTH= 7, ! Month

```

IDATE1 = 17,      ! Date
IYEAR1 = 87,      ! Year

! Surface layer and soil

NISFCL = 1,      ! Flag for the surface layer parameter
                ! (1-prognostic soil model, 0-specified gradients)

! if (NISFCL=1) then
NNZG = 4,        ! The number of vertical levels in the soil model
ZROUGH = .10,    ! Roughness length for land areas
SEATMP = 283.,   ! Water surface temperature for water areas
SOILDZ = 0.,     ! Constant vertical spacing of soil model
                ! (if = 0, use SLZ)
SLZ = -.20, -.05, -.01, 0., ! Variable vertical spacing of soil model
SLMSTR = .8, .7, .4, .3,   ! Initial soil moisture profile in fraction of saturation
! if (NISFCL=0) then
DTHCON = 0.,      ! If no soil model constant potential
                ! temperature gradient between air and
                ! soil
DRTCON = 0.,      ! Constant vapor mixing ratio gradient

```

\$END

\$SOUNDING ! The following data are ignored by the program

```

!-----
! Variables to change for BOTH models
!-----
! Sounding specification if input SNDFILE is not used.
! make sure sounding goes well above
! expected convective cloud top

! Flags for how sounding is specified

IPRSFL=0,      ! 0 - pressure (mb), 1 - heights (m), PRS(1)=sfc press(mb).
ITMPFL=0,      ! 0 - temp(C), 1 - temp(K), 2 - pot. temp(K)
IRH0FL=0,      ! 0 - dew pnt.(C), 1 - dew pnt.(K), 2 - mix rat(g/kg)
                ! 3 - relative humidity in %, 4 - dew pnt depression(K)
IUUVFLG=0,     ! 0 - u,v component(m/s), 1 - umoms-direction, vmoms-speed

PRS=1000., 500., 1000., 2000., 3000., 4000., 6000., 10000., 15000.,

TS= 295., 298., 300., 310., 315., 320., 330., 350., 420.,

RH0= 14.0, 14.0, 12.4, 8.0, 6.0, 2.0, 1.0, .05, .00,

UMOMS= -5., -5., -5., -5., -5., -5., -5., -5., -5.,
! UMOMS= 0., 0., 0., 0., 0., 0., 0., 0., 0.,

VMOMS= 0., 0., 0., 0., 0., 0., 0., 0.,

```

\$END

\$PRNT

! Variables to change for BOTH models

NPLT = 6, ! Number of fields to be printed at each time
! for various cross-sections (limit of 50)

IPLFLD = 'U','V','W',
PP','THETA','RT', ! Field names - see table below

PLFMT(3) = '2PF6.1', ! Format specification if default is unacceptable

IPLTYP = 'BO','BO','BO',
'BO','BO','BO', ! Print/plot type ('PRINT','PLOT','BOTH')
! (first 2 characters only are checked)

IPLVECT= 1, ! Vector flag. If greater than 0, velocity
! vectors will be overlaid on contour plot.
! 1-vectors at every grid point, 2-every other
! point, etc.

IXSCTN = 6*3, ! Cross-section type (1=XZ, 2=YZ, 3=XY)

ISBVAL = 6*4, ! Grid-point slab value for third direction

IAA = 6*-1, ! Window offsets for prints or plots:
IAB = 6*-1, ! left, right, bottom, top respectively
JOA = 6*-1, ! must be .LE. 0
JOB = 6*-1, !

\$END

C	'U'	- UP(M/S)			
C	'V'	- VP(M/S)		'TG'	- TG(K)
C	'W'	- WP(M/S)		'SLM'	- SLM(PCT)
C	'PP'	- PRS(MB)		'CONPR'	- CON RATE
C	'THETA'	- THETA(K)		'CONP'	- CON PCP
C	'THVP'	- THV(K)		'CONH'	- CON HEAT
C	'TV'	- TV(K)		'CONM'	- CON MOIS
C	'RT'	- RT(G/KG)			
C	'RV'	- RV(G/KG)		'RTP'	- RT' G/KG

Appendix B: 'VIEW_PLOT' input job file

\$PRNT

NPLT = 13,	! Number of fields to be printed at each time ! for various cross-sections (limit of 50)
IPLFLD = 13*'W',	! Field names - see table below
PLFMT(3) = '2PF6.1',	! Format specification if default is unacceptable
IPLTYP = 13*'PL',	! Print/plot type ('PRINT','PLOT','BOTH') ! (first 2 characters only are checked)
IPLVECT= 13*1,	! Vector flag. If greater than 0, velocity ! vectors will be overlaid on contour plot. ! 1-vectors at every grid point, 2-every other ! point, etc.
IXSCTN = 4*1,4*2,5*3	! Cross-section type (1=XZ, 2=YZ, 3=XY)
ISBVAL = 1*3,1*4,1*5,1*6,1*7	! Grid-point slab value, third direction
IAA = 13*-1,	! Window offsets for prints or plots:
IAB = 13*-1,	! left, right, bottom, top respectively
JOA = 13*-1,	! must be .LE. 0
JOB = 13*-1,	!

\$END

\$OUTPUT_DATA

TIME_INPUT = 3600.,	! time to print out field
ICOLOR_PLOT = 0,	! 1 for color plot. 0 for black and white.

\$END

Appendix C: The 'PRNT' section control file of the GL-3D job control file

Example 1: with six dynamics fields output

```
$PRNT
!-----
! Variables to change for BOTH models
!-----

      NPLT = 6,           ! Number of fields to be printed at each time
                        ! for various cross-sections (limit of 50)

      IPLFLD = 'U','V','W',
              'PP','THETA','RT', ! Field names - see table below

      PLFMT(3) = '2PF6.1', ! Format specification if default is unacceptable

      IPLTYP = 'BO','BO','BO',
              'BO','BO','BO', ! Print/plot type ('PRINT','PLOT','BOTH')
                        ! (first 2 characters only are checked)

      IPLVECT= 1,         ! Vector flag. If greater than 0, velocity
                        ! vectors will be overlaid on contour plot.
                        ! 1-vectors at every grid point, 2-every other
                        ! point, etc.

      IXSCTN = 6*3,       ! Cross-section type (1=XZ, 2=YZ, 3=XY)

      ISBVAL = 6*4,       ! Grid-point slab value for third direction

      IAA = 6*-1,         ! Window offsets for prints or plots:
      IAB = 6*-1,         ! left, right, bottom, top respectively
      JOA = 6*-1,         ! must be .LE. \0
      JOB = 6*-1,         !

$END
```

Example 2: with only one dynamics field output

```
$PRNT
!-----
! Variables to change for BOTH models
!-----

      NPLT = 1,           ! Number of fields to be printed at each time
                        ! for various cross-sections (limit of 50)

      IPLFLD = 'W',       ! Field names - see table below

      PLFMT(3) = '2PF6.1', ! Format specification if default is unacceptable
```

IPLTYP = 'PL',	! Print/plot type ('PRINT','PLOT','BOTH')
IPLVECT= 0,	! Vector flag. If greater than 0, velocity ! vectors will be overlaid on contour plot. ! 1-vectors at every grid point, 2-every other ! point, etc.
IXSCTN = 1*3,	! Cross-section type (1=XZ, 2=YZ, 3=XY)
ISBVAL = 1*3,	! Grid-point slab value for third direction
IAA = 1*-1,	! Window offsets for prints or plots:
IAB = 1*-1,	! left, right, bottom, top respectively
JOA = 1*-1,	! must be .LE. 0
JOB = 1*-1,	!

\$END

Appendix D: NEPH DATA BASE (NDB) Library

All modules listed below are located in the directory -- NEF_NDB
and contained in the library -- NEF_NDB:NDB.OLB

NOTES: * First call must be to NDB_OPEN.

* All calls to NDB functions must be preceded by a call to
NDB_OPEN.

* When linking in any of these routines the NOPT option file must be
included in the LINK command stream:
\$LINK file_name,...,NOPT/OPT

* The NEPH data dictionary structure (DD) is defined in
NEF_INCLUDES:DD.INC for FORTRAN programs and in
NEF_INCLUDES:DD.H for C programs.

ADD_ENTRY

ISTAT = NDB_ADD_ENTRY(DD).

Add an entry to the NEPH data dictionary and create an output file.
The name of the output file is returned in DD.FILE_NAME. Returns
record length of created output file if success, 0 for failure.

DD (STRUCTURE DD, by reference)

A filled data dictionary structure that is used to generate the
DD entry.

ARCHIVE

ISTAT = NDB_ARCHIVE(TYPE, ENTRY, INIT)

Write the data file(s) associated with the specified TYPE and ENTRY
to tape. Optionally initialize a new archive tape. Tape must
be mounted foreign and a logical called TAPE assigned to the tape drive.
If TAPE is not assigned it will default to ARCBOT\$MUB0: the 8mm drive.
Return 1 for success, 0 for failure.

TYPE (I*4, by value)

Data type id (i.e., DD.TYPE).

ENTRY (I*4, by value)

DD entry number (i.e., DD.ENTRY).

INIT (I*4, by value)

Tape initialization flag: 1-initialize new tape, 0-previously
initialized tape.

CLEAR_ENTRY

ISTAT = NDB_CLEAR_ENTRY(INDX)

Delete the specified entry in the data dictionary bitmap. Note that this routine does not remove an entry from the data dictionary file but simply marks the slot as available so that it can be overwritten at any time. Returns 1 for success, 0 for failure.

INDX (I*4, by value)
Entry number to delete.

COMPARE

ISTAT = NDB_COMPARE(DD1, DD2)

Compares two data dictionary entries and returns a 1 if they are the same and 0 if not. Excluded fields are TYPE, ENTRY, VERSION, FILE NAME, and ID of ancillary file (e.g., EPHEM_ID, TIME_ID).

DD1 (STRUCTURE DD, by reference)
First data dictionary entry to compare.
DD2 (STRUCTURE DD, by reference)
Second data dictionary entry to compare.

COUNT_BOX

CNT = NDB_COUNT_BOX(MASK)

Return the number of bits set in the input data dictionary NEPH box mask. Corresponds to the number of NEPH boxes contained in a data file.

MASK (Array of 2 I*4, by reference)
NEPH box mask contained in data dictionary (DD.BOX).

DELETE_ENTRY

CALL NDB_DELETE_ENTRY(TYPE, ENTRY)

Mark an entry in the data dictionary as off-line and delete the data file; first checks that file has been archived on tape. Returns 1 for success, 0 for failure.

TYPE (I*4, by value)
Type code of entry to delete.
ENTRY (I*4, by value)
Entry number to delete.

FIND_BOX

OFFSET = NDB_FIND_BOX(MASK, BOX)

Determine if specified NEPH box is contained in a data file by testing the data dictionary NEPH box mask. If the box is present return the

offset into the file, if not present return 0.

MASK (Array of 2 I*4, by reference).

NEPH box mask contained in data dictionary (DD.BOX).

BOX (I*4, by value)

NEPH box number to search for.

FIND_EMPTY

ISLOT = NDB_FIND_EMPTY()

Find the first empty slot in the data dictionary, returns entry number when successful, 0 for failure.

GET_DD

COUNT = NDB_GET_DD(TYPE, ENTRY, DD, SIZE)

Get specified data dictionary entries and return them along with a record count. Specify entries by type and entry number. Returns number of entries located when successful, 0 for failure.

TYPE (I*4, by value)

Data type ID, if TYPE=0 return all entries in the DD.

ENTRY (I*4, by value)

Entry number, if ENTRY=0 return all entries for specified type.

DD (STRUCTURE DD, by reference)

An array of empty data dictionary structures that will contain all entries that match the TYPE and ENTRY specifications upon return.

SIZE (I*4, by value)

The size of the DD array.

KILL_ENTRY

CALL NDB_KILL_ENTRY(TYPE, ENTRY)

Permanently remove an entry from the data dictionary and delete the data file. **DO NOT USE THIS ROUTINE** unless you really understand the consequences of this action are. At least talk to Gary before using it. To temporarily remove a file from the disk use NDB_DELETE_ENTRY. Returns 1 for success, 0 for failure.

TYPE (I*4, by value)

Type code of entry to delete.

ENTRY (I*4, by value)

Entry number to delete.

LIST_ENTRY

ISTAT = NDB_LIST_ENTRY(TYPE, ENTRY)

List all entries that match the specified type and entry number. Return 1 for success, 0 for no match or failure.

TYPE (I*4, by value)

Data type ID, if TYPE=0 list all entries in the DD.

ENTRY (I*4, by value)

Entry number, if ENTRY=0 list all entries for specified type.

MAP_DDMAP

ADR = NDB_MAP_DDMAP()

Map the data dictionary bitmap file into memory. Return the memory address of the bitmap.

MAP_NEPHBOX

ISTAT = NDB_MAP_NEPHBOX(DD, ARRAY, BOX, RW_FLAG)

Map 1 NEPH box worth of data for specified type and entry into specified array. Assumes that array is page aligned. Returns 1 for success, 0 for failure.

DD (STRUCTURE /DD/, by reference)

Data dictionary structure containing entry describing data file to be mapped.

ARRAY (I*(*), by reference)

Array to be mapped to data file, must be large enough to contain one NEPH box worth of data, MUST BE PAGE ALIGNED.

BOX (I*4, by value)

NEPH box number to be mapped into memory.

RW_FLAG (I*4, by value)

Read/write flag, 0-readonly 1-read/write

MAPD_NEPHBOX

NBYTES = NDB_MAPD_NEPHBOX(DD, ADDR, BOX, RW_FLAG)

Map 1 Neph box worth of data for specified type and entry into first available memory location. Returns the number of bytes mapped or 0 for failure.

DD (STRUCTURE /DD/, by reference)

Data dictionary structure containing entry describing data file to be mapped.

ADDR (I*4, by reference)

Returned as the section address, the easiest way to use this value is to treat it as a pointer. This means that when calling from a C program the pointer must be passed to the subroutine BY REFERENCE, e.g.

```
int *ptr;
```

```
nbyte = ndb_mapd_nephbox(&dd, &ptr, box);
```

from FORTRAN it is treated as a normal I*4 variable but must be passed by value to a subroutine in order to be used as an array, e.g.

```
integer*4 ptr  
nbyte = ndb_mapd_nephbox(dd, ptr, box)  
call sub1(%val(ptr))  
.  
.  
end
```

```
subroutine sub1(ptr)  
byte ptr(1)  
.  
.
```

BOX (I*4, by value)
NEPH box number to be mapped into memory.
RW_FLAG (I*4, by value)
Read/write flag, 0-readonly 1-read/write

MODIFY_ENTRY

ISTAT = NDB_MODIFY_ENTRY(DD)

Modify the specified existing data dictionary entry to the values contained in the input parameter. Returns 1 for success, 0 for failure.

DD (STRUCTURE /DD/, by reference)
Data dictionary structure containing new values to be placed in the DD entry pointed to by DD.TYPE and DD.ENTRY.

OPEN

ISTAT = NDB_OPEN(CASE_DAY)

Open data base manager for specified case study day. MUST BE CALLED BEFORE CALLS TO ANY OTHER NDB FUNCTIONS. Returns 1 for success, 0 for failure.

CASE_DAY (I*4, by value)
Julian date of selected case study day (e.g., 82162).

OPEN_DD

LUN = NDB_OPEN_DD(RW_FLAG)

Open the data dictionary file for either read or read/write access. Returns the logical unit number assigned to the file or (-1) for failure.

RW_FLAG (I*4, by value)
Access flag: 0-read only, 1-read/write

RECL

LRECL = NDB_RECL(DD)

Compute data file record length such that one logical record contains one Neph box worth of data. Returns record length or 0 for failure.

DD (STRUCTURE DD, by reference)
Data dictionary entry describing data file.

RESTORE

ISTAT = NDB_RESTORE(TYPE, ENTRY)

Restore data file from tape to disk. Use DD entry to locate and validate tape and file numbers. Specify file by type and entry numbers. Return 1 for success, 0 for failure.

TYPE (I*4, by value)
Data type ID (i.e., DD.TYPE).
ENTRY (I*4, by value)
DD entry number (i.e., DD.ENTRY).

SET_BOX

CALL NDB_SET_BOX(MASK, BOX)

Set proper bit in the data dictionary NEPH box mask to indicate that the specified NEPH box is contained in a data file. Returns 0 if box previously set, 1 otherwise.

MASK (Array of 2 I*4, by reference)
NEPH box mask contained in data dictionary (DD.BOX).
BOX (I*4, by value)
NEPH box number to set.

SHOW_BOX

CALL NDB_SHOW_BOX(MASK)

Generate an 8x8 map of available NEPH boxes for the specified DD NEPH box mask.

MASK (Array of 2 I*4, by reference)
NEPH box mask contained in data dictionary (DD.BOX).

SHOW_DD

CALL NDB_SHOW_DD(DD)

Generate a formatted output to SYSS\$OUTPUT of the specified DD entry, format depends on data type (i.e., DD.TYPE).

DD (STRUCTURE DD, by reference)
Data dictionary structure containing the entry to be output.

TYPE

`LABEL/ID = NDB_TYPE(TYPE)`

Returns a 4-character type label for a specified numeric data type ID, or returns a numeric data type id for a specified 4 character type label. Returns 0 for unknown label or type.

TYPE (I*2, by reference)

Data type ID [i.e.,	INTEGER*4 LAB	]
	[	INTEGER*2 TYPE
	[	LAB = NDB_TYPE(TYPE)
	[	TYPE '(A4)', LAB

 ...OR

TYPE (C*4, by reference)

Data type label [i.e.,	INTEGER*4 ID	]
	[	CHARACTER *4 LBL/SGDB/
	[	ID = NDB_TYPE (%REF(LBL))
	[	TYPE '(I4)', ID

UNMAP_NEPHBOX

`ISTAT = NDB_UNMAP_NEPHBOX(DD)`

Unmaps memory from data file associated with specified data dictionary entry. Assumes that the memory area was mapped using either `NDB_MAP_NEPHBOX` or `NDB_MAPD_NEPHBOX`. Frees file to be deleted or mapped to another memory section. Returns 1 for success, 0 for failure.

DD (STRUCTURE DD, by reference)

A filled data dictionary structure that contains the DD entry for the file that is to be unmapped.

Appendix E: NDB data base utility

The Neph Data Base (NDB) utility command allows interactive query of the data base by Type and Entry number. Provides a quick look at the data through contour analyses and image displays on appropriate data types. Also used for archive and restore, deletion, and display of individual entries when appropriate qualifiers are selected. For each operation a case study day (YYDDD) must be selected; if no day is specified the previous specification is remembered.

Default operation is to produce a screen listing of the Data Dictionary entry for the specified data Type and Entry. Multiple entries can be selected by using wildcard options (0) for either Type and/or Entry.

Format:

NDB/[qualifiers] [type] [entry]

Parameters:

Type

Specifies the data type that is to be acted upon. Data type can be specified by either its numeric code or its 4-character abbreviation. Use NDB/TYPE to get a listing of code and matching abbreviations. Type will accept "0" as a wildcard type; this will match all types in the data base. The default value is 0.

Entry

Specifies a unique entry number in the Data Dictionary. Type and Entry together uniquely identify every entry in the Data Dictionary. Entry will accept "0" as a wildcard entry; this will match all entry numbers in the data base. The default value is 0.

Command_Qualifiers:

/BOX=Neph_box_number

Display an image representation of the selected data file for the specified Neph box number (1-64) on the Adage monitor. Currently this routine only works for data Type SGDB (code=1).

/CASE=YYDDD

Specifies the case study day for this data base transaction. The case study day must be selected on the first call to NDB. Subsequent calls will use the previously specified value until a new day is specified.

/CONTOUR=Neph_box_number

Produce a contour plot of the selected data file for the specified Neph box (1-64) on any NCAR supported graphics device. The facility is used for a quick look at the data to help detect any gross problems. It uses the fast NCAR analysis and contour routine.

/DAYS

Generate a list of available case study days.

/DELETE

Remove the data file associated with the specified Type and Entry from the online disk directory. This routine will first check that the file has been previously archived on tape. Interactive verification is required before each file is deleted.

This option can be used in combination with /ARCHIVE to perform both operations together.

/FULL

Gives the full Data Dictionary entry listing for the specified Type and Entry including the Neph box map.

/MAP=Neph_box_number

Generate a map of geopolitical outlines for the specified Neph box number (1-64) on any NCAR supported graphics device.

/RESTORE

Restore the data file associated with the specified Type and Entry from an archive tape and put back in the online disk directory. Entries that are archived but not currently on disk are designated with an "*" following the archive information in the NDB listing of the Data Dictionary entry.

The tape drive designation (e.g. MUB0:) must be mounted /FOREIGN and assigned to a logical called "TAPE" prior to selecting this option. If no logical assignment is made, the routine defaults to drive MUB0: on ARCUNO (the 8mm drive).

/TYPE

Generate a listing of numeric data Type codes and the matching Type abbreviations.

/ARCHIVE

Writes the data file(s) associated with the specified Type and Entry to tape in a standard NDB format. Type and Entry wildcard characters can be used to archive more than one file at a time. Interactive verification is required before each entry is archived. Archive tape and file number are stored in the archive section of the Data Dictionary entry.

The tape drive designation (e.g. MUB0:) must be mounted /FOREIGN and assigned to a logical called "TAPE" prior to selecting this option. If no logical assignment is made, the routine defaults to drive MUB0: on ARCUNO (the 8mm drive).

Standard NDB format includes the Data Dictionary entry followed by the data file in the same format as it was stored on disk.

/INITIALIZE

Initializes a new archive tape to the NDB standard format. Must be used when a tape is used for the first time.

CAUTION: this will wipe out any information that was previously written to the tape.

/NEW

Archive only the subset of the specified entries that have not been previously archived.

Examples:

1. \$NDB/CASE=85010

Generate a listing of the Data Dictionary entries for case study day 85010. Results of this entry look like this:

Case Study Day - 85010

Entry	Type	Mesh	Len	Ver	Date	Time	Channel	Bits	ID	Entry	Tape/File
3	SGDB	64	1	1			OLS Vis	6		1	1/010
4	SGDB	64	1	1			OLS IR	6		2	1/011
1	Ephm	4	16	1			OLS Vis				1/008
2	Ephm	4	16	1			OLS IR				1/009
5	SfcT	8	2	1	85009	2100				6	1/032
11	SfcT	8	2	1	85010	0000				13	1/033
14	SfcT	8	2	1	85010	0300*				15	1/034
16	SfcT	8	2	1	85009	1200				17	1/035
20	SfcT	8	2	1	85009	1500				21	1/036
22	SfcT	8	2	1	85009	1800				23	1/037
7	Geog	8	1	1							1/012
8	Terr	8	2	1							1/013
12	UprA	1	40	1	85010	1200					0/000
6	TTim	0	12	1	85009	2100					1/038
13	TTim	0	12	1	85010	0000					1/039
15	TTim	0	12	1	85010	0300					1/040
17	TTim	0	12	1	85009	1200					1/041
21	TTim	0	12	1	85009	1500					1/042
23	TTim	0	12	1	85009	1800					1/043

Where ENTRY and TYPE are as specified, MESH is in fractions of a whole mesh box (0 indicates Neph box level), LEN is the length in bytes of each data point, VER is version number, DATE and TIME are given for data types where it is appropriate, SATELLITE information is also given where appropriate, ALT ENTRY indicates the entry number of a supporting data file or, in the case of output files (TYPE > 1000), the input control structure, and ARCHIVE indicates tape and file number. An * next to time information indicates a forecasted field. An * next to archive information indicates the file is not stored in an online directory but must be restored from tape.

2. \$MOUNT/FOREIGN MSA0: \$DEFINE TAPE MSA0: \$NDB/ARCHIVE/DELETE 1 0

Backup all type 1 (SGDB) data to the tape currently loaded on drive MSA0:. Use NDB archive format. After the archive operation is successfully completed the file is removed from the online directory.

3. \$MOUNT/FOREIGN MUB0:
\$NDB/RESTORE/CASE=84285 GEOG 8

Restore the Geography (Type=6) data file associated with entry number 8 from the archive tape mounted on drive ARCUNO\$MUB0: (default) to the 84285 case study directory. Compatibility checking will be performed to insure that the mounted tape is the correct one containing the archive of the specified data file.

4. \$NDB/TYPE

Generate a list of the possible data type codes and the matching abbreviations. Looks like:

ID	TYPE
1	-- SGDB
2	-- Ephm
3	-- SfcT
4	-- UA_T
5	-- UA_H
6	-- Geog
7	-- Terr
8	-- BkBt
9	-- UprA
10	-- TTim
1001	-- RGS
1002	-- Oput

Appendix F: AVHRR Mapping Utility User Guide

The Mapping Utility reformats AVHRR GAC data from raw scan line format to the RTNEPH grid system according to the SGDB standard. It is described in detail in Section 5 of the report. This User Guide provides detailed instructions on using the mapping utility. The utility produces a set of six RTNEPH grids. Five of the grids contain reformatted imagery data from the five AVHRR sensor channels. The sixth grid contains reformatted ephemeris data. The results of the utility are stored in the Nephanalysis Data Base (NDB). The gridded satellite imagery can be displayed using the AIMS ADAGE NDB display software.

1) Unpacking the AVHRR Data

The Mapping Utility was written to process data obtained on tape from NOAA NESDIS in packed binary format. Prior to running the mapping utility, the AVHRR data to be mapped must be unpacked from tape. This is accomplished using the GL AVHRR unpacking utility. First, a directory must be created to hold the unpacked data as follows:

USER\$DISK_20:[NEPH.DATA.TAPE##] where ## is a two-digit tape
identification number assigned
by the user.

USER\$DISK_20:[TAPE##] must be the working directory when the unpacking utility is run.

Due to the large amount of mass storage needed to store the unpacked data (approximately 2.8 Mbytes) the storage disk is hardcoded in the mapping utility program as USER\$DISK_20. At the user's discretion, this can be changed. To run the unpacking utility type:

@USER\$DISK_23:[BOB]AVHRR.COM.

You will be prompted for "unpack" or "display". Enter "unpack".

The unpacking utility unpacks a maximum of 5120 scan lines to save memory. The following files will be created:

Ch1_alb.dat	Normalized Albedoes
Ch2_alb.dat	
Ch3_temp.dat	Brightness Temperatures
Ch4_temp.dat	
Ch5_temp.dat	
All_loc.dat	Earth Locations
All_solzen.dat	Solar Zenith Angles
All_time.dat	Scan Line Times

2) Determining the Scan Coverage - WORLD_MAP

The user must determine which Neph boxes should be processed for a given set of data. The appropriate combination of Neph boxes and a specific range of 500 scan lines of data can be selected using WORLD_MAP. WORLD_MAP is a program which graphically displays AVHRR data on a polar map in scan line format (See Figure 5.6). The source code is found in USER\$DISK_1:[SPARROW.AVHRR]WORLD_MAP.FOR. The executable is WORLD_MAP.EXE.

WORLD_MAP prompts for four things:

- GKS terminal type - the prompt is explained in the program,
- directory name - where satellite data to be displayed is stored
Ex. User\$disk_20:[neph.data.tape##],
- first scan line to be displayed - unpacked data files contain up to 5120 scan lines of data. The user chooses the first line which is to be displayed,
- number of scan lines to be displayed - up to 500 scan lines can be displayed at one time, and
- scan line increment - every i'th scan line will be displayed.

3) RUNNING THE MAPPING UTILITY - MAP

The mapping utility is located in USER\$DISK_1:[SPARROW.AVHRR]MAP.C. To run the utility type "Map". The program will prompt for the following inputs:

- tape - two-digit tape number which identifies the directory where the AVHRR data is stored.
Example:
for user\$disk_20[neph.data.tape01],
enter '01'
- numbox - number of individual Neph boxes which are to be processed in this run. This number cannot exceed 4. This limit exists because it is rare that 500 scan lines will cover more than 4 Neph boxes.
- box[i] - the program will prompt for individual box numbers based on "numbox".

***** IMPORTANT NOTE *****

The boxes processed in a single run must either all be between 1 and 32 or all be between 33 and 64.

Correct Example: box[1] = 17

box[2] = 19
box[3] = 29

Incorrect Example: box[1] = 17
box[2] = 19
box[3] = 33 <---

This is required due to problems in the satellite location data caused by crossing the date line. The Neph-boxes processed in one run must all be either east or west of the date line.
(See Sections 5.5 and 5.7)

- JMIN - the starting row of AVHRR data to be processed.
Exactly 500 scan lines will be processed in one run. Since there are 5120 scan lines per AVHRR data set, if JMIN is > 4619, JMIN defaults to 4619.

Actual CPU time to run the Mapping Utility may be anywhere between 20 minutes and 6 hours to process a single box. Long run times are the result of certain attributes of the AVHRR location data; in some cases the binary search routine has difficulty narrowing down the location of the nearest neighbor. Sometimes, a nearest neighbor point is not found, even though points are found for all the surrounding grid points. This often occurs along the boundaries of the search boxes. Because the grids are updated only if an entire quarter-mesh box can be completed, this leaves a hole in the mapped grids. These holes will usually be filled in later runs of the utility with data from overlapping orbits. If the run times are very long, over 3 hours per box, it may be useful to more closely examine the data using WORLD_MAP and verify no problems exist in the data. Refer to Section 5.8 for a discussion of potential problems.