Final REPORT

Cloud Assimilation into the Weather Research and Forecast (WRF) Model

Project Period of Performance
February 4, 2010 – February 28, 2011

Prepared for:

Bright Dornblaser
Texas Commission on Environmental Quality (TCEQ)

Submitted to

LAMAR UNIVERSITY
Thomas C. Ho
Department of Chemical Engineering, Lamar University
P. O. Box 10053, Beaumont, Texas 77710

Prepared by:

Arastoo Pour Biazar, Kevin Doty, Yun-Hee Park
National Space and Technology Center
University of Alabama – Huntsville

Richard T. McNider
Southeastern Atmospheric Modeling
Huntsville,AL

February 28, 2011

Table of Contents

Table of Contents i

Table of Figures: v

List of Tables: x

1 SUMMARY 1

2 INTRODUCTION 5

3 BASELINE SIMULATION 7

3.1 Model Configuration 7

3.2 Establishing the Baseline Simulation 8

3.3 Comparison with MM5 and Weather Charts 8

3.4 METSTAT Evaluation 11

3.5 Evaluating Cloud Prediction Using GOES Observations 20

3.6 Conclusions from Baseline Simulation 27

4 PORTING MM5 IMPLEMENTATION TO WRF 29

4.1 Overall Summary of Convective Suppression Run 30

4.1.1 Time Series of Agreement Index 34

4.1.2 Plots for 1800 UTC 9 July 2006 36

4.1.3 Conclusions 38

4.2 One-Dimensional Variation Technique 39

5 ALTERNATE APPROACH 41

5.1 Purpose and Strategy 41

5.2 Adjustment of the Model Environment to Support Clouds 41

5.3 Development of Cloud and Satellite Relationships 43

5.4 Relationships Between Model Variables and Cloud Water in the Model 45

5.5 Alternative Threshold Relations 52

5.6 Use of Model Variables to Develop Satellite Contingency Tables 55

5.7 Monthly Statistics 60

5.8 Implementation in WRF 61

6 Summary and Conclusions 67

References 69

APPENDIX A **71**

1 Introduction to Convective Suppression **73**

2 GOES Data **73**

3 Overview of Changes **74**

3.1 WPS Modifications *74*

3.2 WRF Modifications *78*

3.3 Changes to the GOES Regridding Software *79*

4 Data Flow **81**

5 Implementation **83**

5.1 Simple Description of Implementation Steps *84*

5.2 Discussion of Implementation Steps *84*

6 Application Example **86**

6.1 Statistical Tools *87*

6.2 Analysis of Test Run *88*

7 References **93**

APPENDIX B **95**

1 Introduction **97**

2 Brief Summary of WDIAG Module **97**

3 Namelist File "wdiag_options.inp" Options **98**

3.1 Grid, interpolation, and analysis options *99*

3.2 Parameters used when hz_interp_type_out = 'GAUSSIAN' *101*

3.3 Parameters for analytical w target *101*

3.4 Parameters for Sequential Over-Relaxation *102*

3.5 Parameters used in subroutine check_vertical_velocities *103*

4 Arguments to WDIAG Driver Subroutine **103**

5 Basic Variational Equations **108**

6 Development of O'Brien (1970) Equations **110**

7 Application of O'Brien Equations to Sigma-h Coordinates **114**

8 Divergence Adjustment Details **117**

9 Divergence Adjustment Calculations **119**

9.1 Grid Considerations *119*

9.2 Basic Adjustment Steps 120

10 WDIAG Flow Charts 121

11 REFERENCES 139

Table of Figures:

FIGURE 3.1. SURFACE ANALYSIS (TOP), MM5 SIMULATION (LOWER LEFT), AND WRF SIMULATION (LOWER RIGHT) RESULTS FOR 21 GMT, AUGUST 19, 2006 WHERE A STATIONARY FRONT EXTENDING FROM SOUTHWEST TO NORTHEAST CONTINENTAL U.S. WAS PRESENT. 10
FIGURE 3.2. SURFACE ANALYSIS (TOP), MM5 SIMULATION (LOWER LEFT), AND WRF SIMULATION (LOWER RIGHT) RESULTS FOR 20 GMT, AUGUST 20, 2006. 11
FIGURE 3.3. TEMPERATURE STATISTICS FOR THE FIRST PART OF AUGUST 2006 FROM WRF SIMULATIONS USING KAIN-FRITSCH, GRELL-DEVENYI ENSEMBLE, AND NEW GRELL SCHEME FOR CONVECTIVE PARAMETERIZATION. 13
FIGURE 3.4. MIXING RATIO (G/KG) STATISTICS FOR THE FIRST PART OF AUGUST 2006 FROM WRF SIMULATIONS USING KAIN-FRITSCH, GRELL-DEVENYI ENSEMBLE, AND NEW GRELL SCHEME FOR CONVECTIVE PARAMETERIZATION. 14
FIGURE 3.5. WIND SPEED STATISTICS FOR THE FIRST PART OF AUGUST 2006 FROM WRF SIMULATIONS USING KAIN-FRITSCH, GRELL-DEVENYI ENSEMBLE, AND NEW GRELL SCHEME FOR CONVECTIVE PARAMETERIZATION. 15
FIGURE 3.6. LOCATION OF WEATHER STATIONS IN TEXAS THAT WAS USED IN THE ANALYSIS. 16
FIGURE 3.7. AVERAGE TIME SERIES FOR TEMPERATURE, MIXING RATIO, AND WIND SPEED DURING AUGUST 18-23, 2006 OVER STATIONS IN TEXAS. THE RESULTS FROM WRF SIMULATIONS USING KAIN-FRITSCH, GRELL-DEVENYI ENSEMBLE, AND NEW GRELL SCHEME FOR CONVECTIVE PARAMETERIZATION ARE PLOTTED OVER SURFACE OBSERVATIONS (OPEN CIRCLES). 17
FIGURE 3.8. TIME SERIES FOR TEMPERATURE, MIXING RATIO, AND WIND SPEED DURING AUGUST 18-23, 2006 OVER A STATION IN SOUTHWEST TEXAS. THE RESULTS FROM WRF SIMULATIONS USING KAIN-FRITSCH, GRELL-DEVENYI ENSEMBLE, AND NEW GRELL SCHEME FOR CONVECTIVE PARAMETERIZATION ARE PLOTTED OVER SURFACE OBSERVATIONS (OPEN CIRCLES). 18
FIGURE 3.9. TIME SERIES FOR WIND SPEED DURING AUGUST 18-23, 2006 OVER STATIONS IN NORTH TEXAS. THE LOCATION OF THE STATIONS ARE MARKED IN FIGURE 7. THE RESULTS FROM WRF SIMULATIONS USING KAIN-FRITSCH CONVECTIVE PARAMETERIZATION TOGETHER WITH LIN OR WSM6 MICROPHYSICS OPTIONS ARE PLOTTED OVER THE SURFACE OBSERVATIONS (OPEN CIRCLES). 19
FIGURE 3.10. TIME SERIES FOR WIND SPEED DURING AUGUST 18-23, 2006 OVER STATIONS IN SOUTH TEXAS. THE LOCATION OF THE STATIONS ARE MARKED IN FIGURE 7. THE RESULTS FROM WRF SIMULATIONS USING KAIN-FRITSCH CONVECTIVE PARAMETERIZATION TOGETHER WITH LIN OR WSM6 MICROPHYSICS OPTIONS ARE PLOTTED OVER THE SURFACE OBSERVATIONS (OPEN CIRCLES). 20
FIGURE 3.11. OBSERVED SATELLITE CLOUD VERSUS MODEL CLOUD FOR 20 GMT, AUGUST 16, 2006. CLOUD INDEX FOR A) GOES OBSERVATION, B) WRF MODEL SIMULATION USING KAIN FRITSCH CONVECTIVE PARAMETERIZATION, AND C) COLOR CODED AREAS OF AGREEMENT (RED AND BLUE) AND DISAGREEMENT (GREEN FOR OVER-PREDICTION, AND ORANGE FOR UNDER-PREDICTION) BETWEEN MODEL AND OBSERVATION ARE PRESENTED IN THE FIGURE. THIS CASE HAS ALMOST PERFECT AGREEMENT IN THE FRACTIONAL CLOUD/CLEAR AREA, WITH WRF SHOWING 38% OF THE AREA BEING CLOUDY VERSUS 36% FOR GOES. BUT THERE IS ONLY ABOUT 70% AGREEMENT WITH THE OBSERVED CLOUDS (SPATIAL DISAGREEMENT BETWEEN GOES AND WRF IS ABOUT 30%). 22
FIGURE 3.12. COLOR CODED AREAS OF AGREEMENT (RED AND BLUE) AND DISAGREEMENT (GREEN FOR OVER-PREDICTION, AND ORANGE FOR UNDER-PREDICTION) BETWEEN MODEL AND GOES OBSERVATION FOR 20 GMT, AUGUST 28, 2006. THE AGREEMENT IS 77% FOR KAIN-FRITSCH (KF) CONVECTIVE PARAMETERIZATION, VERSUS 75% FOR GRELL-DEVENYI (GD) AND 76% FOR NEW GRELL (NG) OPTIONS. THERE IS NO SIGNIFICANT DIFFERENCE BETWEEN GD AND NG FOR THIS SCENE. TABULATED VALUES FOR THE NUMBER OF GRIDS IN AGREEMENT AND DISAGREEMENT ARE ALSO PRESENTED IN THE FIGURE. 23
FIGURE 3.13. TOP PANEL SHOWS THE FRACTIONAL AREA OF THE DOMAIN THAT WAS COVERED BY CLOUDS FOR EACH DAY OF AUGUST FOR GOES AND EACH OF THE WRF SIMULATIONS. THE BOTTOM

PANEL EXHIBITS THE DAILY FRACTIONAL AGREEMENT BETWEEN THE MODEL AND GOES OBSERVATION FOR EACH WRF SIMULATION. 24
FIGURE 3.14. WRF OVER- AND UNDER-PREDICTION OF CLOUDS PRESENTED AS THE FRACTIONAL AREA OF COVERAGE FOR EACH OF THE CONVECTIVE PARAMETERIZATION OPTIONS. IN GENERAL MODEL OVER-PREDICTS CLOUDS IN MORE AREAS THAN UNDER-PREDICTING CLOUDS. GD AND NG SHOW A LARGER GAP BETWEEN THE AREAS UNDER- AND OVER-PREDICTED THAN KF OPTION. 26
FIGURE 4.1. CLOUD TOP PERCENTAGES (Y-AXIS) VERSUS CLOUD TOP HEIGHT CLASSES (X-AXIS) FOR THE FOLLOWING DATA SOURCES: 1). CONTROL WRF RUN (BLUE BARS), 2). THE GOES ASSIMILATION WRF RUN (VIOLET BARS), AND 3). THE GOES OBSERVATIONS (YELLOW BARS). THE CLOUD TOP HEIGHT CLASSES (X-AXIS) ARE IN KM WITH THE LABELS REFERRING TO THE TOP END OF EACH 2-KM CLASS. FOR EXAMPLE, THE 12-KM BIN REPRESENTS CLOUD TOP HEIGHTS GREATER THAN 10-KM BUT LESS THAN OR EQUAL TO 12-KM. THE DATA COVER THE PERIOD FROM 00 UTC 5 JUNE 2006 THROUGH 12 UTC 28 SEPTEMBER 2006. THE SUM OF ALL PERCENTAGES FOR EACH DATA SOURCE SUM TO 100%. 32
FIGURE 4.2. OBSERVED GOES CLOUD TOP HEIGHT PERCENTAGES (Y-AXIS) VERSUS OBSERVED CLOUD TOP HEIGHT CLASSES (X-AXIS) FOR THE GOES ASSIMILATION WRF RUN (VIOLET BARS IN FIG. 1) FOR THOSE GRID POINTS AND TIMES WHEN THE MODEL CLOUD TOP HEIGHTS WERE BETWEEN 10 AND 16 KM. AS IN FIG. 1, THE CLOUD TOP HEIGHT CLASSES (X-AXIS) ARE IN KM WITH THE LABELS REFERRING TO THE TOP END OF EACH 2-KM CLASS. THE "ZERO" X-AXIS CLASS CONTAINS THOSE SITUATIONS WHERE THE MODEL HAD CLOUDS BUT GOES INDICATED CLEAR CONDITIONS. PERCENTAGES ARE RELATIVE TO THE TOTAL NUMBER OF MODEL GRID POINTS MEETING THE 10-16 KM CLOUD TOP CRITERION. 33
FIGURE 4.3. CLOUD TOP PERCENTAGES (Y-AXIS) VERSUS CLOUD TOP HEIGHT CLASSES (X-AXIS) FOR THE CONTROL WRF CONTROL AND GOES ASSIMILATION RUNS ACROSS THE CLOUDY-CLEAR CONTINGENCY TABLE CATEGORIES. THE CLOUD TOP HEIGHT CLASSES (X-AXIS) ARE IN KM WITH THE LABELS REFERRING TO THE TOP END OF EACH 2-KM CLASS. SOLID BARS REFER TO THE CONTROL RUN, WHEREAS SHADED BARS REFER TO THE ASSIMILATION RUN. THE COLORS FOLLOW THE SAME LOGIC AS IN TABLE 1. BLUE REFERS TO CASES WHERE THE MODEL AND GOES HAD CLOUDS. GREEN REFERS TO CASES WHERE THE MODEL HAD CLOUDS BUT GOES WAS CLEAR. ORANGE REFERS TO CASES WHERE THE MODEL WAS CLEAR BUT GOES HAD CLOUDS. THE CASE WHERE BOTH THE MODEL AND GOES HAD CLEAR CONDITIONS IS NOT PLOTTED. THE PERCENTAGE OF CLEAR POINTS FOR THE CONTROL RUN WAS ABOUT 43%, AND ABOUT 42% FOR THE ASSIMILATION RUN. 34
FIGURE 4.4. AGREEMENT INDEX (Y-AXIS) VERSUS JULIAN DAYS (X-AXIS) FOR THE CONTROL WRF RUN (BLUE CURVE) AND THE GOES ASSIMILATION WRF RUN (VIOLET CURVE). TOP PANEL IS FOR JUNE 2006. BOTTOM PANEL IS FOR JULY 2006. VALUES BELOW 0.40 ARE PERIODS WHERE GOES DATA WERE MISSING. DATA FREQUENCY IS HOURLY. 35
FIGURE 4.5. AGREEMENT INDEX (Y-AXIS) VERSUS JULIAN DAYS (X-AXIS) FOR THE CONTROL WRF RUN (BLUE CURVE) AND THE GOES ASSIMILATION WRF RUN (VIOLET CURVE). TOP PANEL IS FOR AUGUST 2006. BOTTOM PANEL IS FOR SEPTEMBER 2006. VALUES BELOW 0.40 ARE PERIODS WHERE GOES DATA WERE MISSING. DATA FREQUENCY IS HOURLY. 36
FIGURE 4.6. PLOTS ARE FOR THE TIME 1800 UTC 9 JULY 2006 FOR THE SIMULATION RUN. TOP PANEL: COLORS CORRESPOND TO THE CONTINGENCY TABLE IN TABLE 2.1. BLUE CELLS ARE WHERE BOTH MODEL AND SATELLITE INDICATED CLOUDS, GREEN CELLS ARE WHERE THE MODEL INDICATED CLOUDS AND THE SATELLITE INDICATED CLEAR CONDITIONS, ORANGE CELLS ARE WHERE THE MODEL HAD CLEAR CONDITIONS AND THE SATELLITE INDICATED CLOUDS, AND RED CELLS ARE WHERE BOTH MODEL AND SATELLITE INDICATED CLEAR CONDITIONS. BOTTOM PANEL: SIMULATION CLOUD TOP HEIGHTS IN KM FOR THE "ORANGE CELLS" IN THE TOP PANEL. 37
FIGURE 4.7. GOES INSOLATION IN W M^{-2} FOR THE TIME 1800 UTC 9 JULY 2006. 38
FIGURE 4.8. CURRENT STATUS OF TWO APPROACHES OF CLOUD ASSIMILATION WORK. COLOR AND BRIEF DESCRIPTION: (BLUE; PATH 1, TRANSITIONING/STREAMLINING MM5 APPROACH TO WPS/WRF SYSTEM), (ORANGE; PATH 2, SIMPLIFIED APPROACH USING THE TOOLS FROM PATH 1), (GREEN; STATISTICAL PACKAGE USED IN BOTH APPROACHES). 39
FIGURE 5.1 SCATTER PLOT OF CLOUD ALBEDO VERSUS TOTAL CLOUD LIQUID WATER 44
FIGURE 5.2 MAP OF THE NATURAL LOG OF MODEL CLOUD WATER. NOTE THIS DEPICTION IS CLOSE TO THAT OF THE RELATIVE CLOUD ALBEDO. 44

FIGURE 5.3 MAPS OF MODEL VERTICAL VELOCITY AND CLOUD ALBEDO FOR A PARTICULAR MODEL TIME. 45
FIGURE 5.4 SCATTER PLOT OF TOTAL CLOUD WATER AND MAXIMUM VERTICAL VELOCITY IN THE MODEL COLUMN. 46
FIGURE 5.5 SCATTER PLOT OF CLOUD ALBEDO VERSUS MAXIMUM VERTICAL VELOCITY IN THE COLUMN. 46
FIGURE 5.6 MAP OF CLOUD ALBEDO AND POINTS SELECTED FOR FURTHER ANALYSIS. SMALL CIRCLE ON MAP SHOW SELECTED POINTS. COORDINATES ARE GIVEN IN FIGURE. 47
FIGURE 5.7 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLEAR POINT (40, 90) 48
FIGURE 5.8 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLEAR POINT (80, 90) 48
FIGURE 5.9 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLOUDY POINT (115, 90) 49
FIGURE 5.10 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLEAR POINT (40, 70) 50
FIGURE 5.11 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLOUDY POINT (60,56) 50
FIGURE 5.12 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLOUDY POINT (110,60) 51
FIGURE 5.13 PROFILES OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLEAR POINT (40,50) 51
FIGURE 5.14 OF VERTICAL VELOCITY (UPPER LEFT), RELATIVE HUMIDITY (UPPER RIGHT) AND TEMPERATURE (LOWER LEFT) FOR CLOUDY POINT (90,50) 52
FIGURE 5.15 GRAPHICAL DEPICTION OF MODEL RELATIONSHIPS. UPPER LEFT SHOWS MODEL CLOUD ALBEDO. UPPER RIGHT SHOWS MODEL VERTICAL VELOCITY. LOWER LEFT SHOWS CLOUDS WITH CONTINGENCY OF MODEL VERTICAL VELOCITY GREATER THAN .02 M/S AND CLOUD ALBEDO >.1. 54
FIGURE 5.16 IMAGES OF MODEL CLOUD ALBEDO AND NEGATIVE VERTICAL VELOCITY. BOTTOM LEFT SHOW COINCIDENCE OF W >0.0 AND CLOUD ALBEDO> .1 54
FIGURE 5.17 MODEL CATEGORIZATION OF RELATIVE HUMIDITY 55
FIGURE 5.18 COMPARISON OF MODEL CLOUDS VERSUS SATELLITE CLOUDS. UPPER LEFT SHOWS MODEL CLOUDS. UPPER RIGHT SHOWS GOES CLOUDS. LOWER LEFT SHOWS MODEL CLOUD DIFFERENCES IN A CLOUD (C) AND NO CLOUD (NC) CATEGORIES GRAPHICALLY AND IN TABULAR STAISTICAL FORM (GRID POINTS). 56
FIGURE 5.19 SAME AS FIGURE 18 EXCEPT FOR USE OF THE GRELL DEVENYI CONVECTIVE PARAMETERIZATION. 57
FIGURE 5.20 IMAGE SHOWING AREAS WHERE MODEL HAS CLOUDS AND GOES DOES NOT. THIS SHOWS WHERE CLOUDS NEED TO BE SUPPRESSED. 57
FIGURE 5.21 DEPICTION OF WHERE MODEL HAS CLOUDS AND GOES DOES NOT. YELLOW PIXELS INDICATE WHERE MODEL W>0.0. GREEN INDICATES WHERE W<0.0. THUS, YELLOW INDICATES WHERE CLEARING CAN PERHAPS BE ACCOMPLISHED BY IMPOSING NEGATIVE W. 58
FIGURE 5.22 DEPICTION OF WHERE MODEL IS CLEAR AND GOES HAS CLOUDS (GREEN PIXELS). SHOW WHERE CLOUDS MUST BE SUPPORTED. 59
FIGURE 5.23 DEPICTION OF WHERE GOES HAS CLOUDS BUT MODEL HAS W<0.0. THE GREEN PIXELS SHOW WHERE THE GOES HAS CLOUDS BUT MODEL W<0.0 . THUS, GREEN PIXELS SHOW WHERE CLOUDS CAN PERHAPS BE CREATED IF W>0 CAN BE IMPOSED. 60
FIGURE 5.24. IMAGE SHOWING AREAS OF AGREEMENT/DISAGREEMENT BETWEEN WRF AND GOES (OBSERVED) CLOUD FIELDS. AREAS OF AGREEMENT ARE DEPICTED IN BLUE (FOR CLEAR SKY) AND RED (FOR CLOUDY SKY) WHILE ORANGE INDICATES UNDER-PREDICTION OF CLOUDS BY WRF AND GREEN INDICATES OVER-PREDICTION OF CLOUDS BY WRF. THE CONTINGENCY TABLE ON THE TOP RIGHT CORNER SHOWS THE NUMBER OF GRID CELLS IN EACH CATEGORY. 63
FIGURE 5.25. SNAPSHOT OF MODIFIED WIND FIELDS FOR 20 GMT, AUGUST 19, 2006. COLOR FILLED CONTOURS REPRESENT VERTICAL VELOCITY. HORIZONTAL WIND FIELD WAS MODIFIED TO SUSTAIN THE TARGET VERTICAL VELOCITY. THE FIGURE ON THE LEFT IS BEFORE THE ADJUSTMENT, WHILE THE FIGURE ON THE RIGHT IS AFTER THE ADJUSTMENT. HIGHLIGHTED AREAS SHOW WHERE SUBSIDENCE IS CHANGE TO LIFTING AND VICE VERSA. 64
FIGURE 5.26. AGREEMENT INDEX (AI) FOR MONTH-LONG WRF SIMULATIONS CONFIGURED WITH NEW GRELL, GRELL-DEVENYI, AND KAIN-FRITSCH CONVECTIVE PARAMETERIZATIONS. ALSO, IN THE

FIGURE ARE THE RESULTS FROM TWO SIMULATIONS WITH CLOUD ASSIMILATION CONFIGURED WITH KAIN-FRITSCH AND GRELL-DEVENYI. 65
FIGURE 4.1 DATA FLOW OF GOES DATA FROM THE PROGRAM METGRID.EXE TO WRF.EXE. BLUE BOXES DESIGNATE READ OPERATIONS, AND ORANGE BOXES DENOTE WRITE OPERATIONS. THE FIRST LINE IN EACH BOX IS A SEQUENTIAL STEP NUMBER. THE SECOND LINE IS AN EXAMPLE OF THE FILENAME IN THE READ OR WRITE OPERATION. THE THIRD LINE IS THE PROGRAM WHICH PERFORMS THE IO FUNCTION. LINES FOUR AND BEYOND GIVE THE VARIABLE NAMES INVOLVED. 82
FIGURE 5.1 FLOWCHART OF THE MAIN STEPS IN THE WPS/WRF PACKAGE WITH ADDITIONAL STEPS REQUIRED TO IMPLEMENT THE PROCESSING OF THE GOES DATA. BLUE BOXES DENOTE STANDARD STEPS WHICH ARE NEEDED REGARDLESS OF WHETHER GOES DATA ARE BEING USED OR NOT. ORANGE BOXES DENOTE STEPS REQUIRED TO IMPLEMENT THE PROCESSING OF THE GOES DATA. STEP NUMBERS ARE IN PARENTHESES. PROGRAM NAMES ARE IN BOLD CASE SUCH AS METGRID.EXE. 85
FIGURE 6.1 MODEL BIAS (TOP) AND FALSE FRACTION INDEX (BOTTOM) FOR THE 72-H 30-KM WRF SIMULATION FOR THE PERIOD 00 UTC 4 FEBRUARY THROUGH 00 UTC 7 FEBRUARY 2008. RUN WITH CONVECTIVE SUPPRESSION OF DEEP AND SHALLOW CONVECTION IN CLEAR AREAS USING GOES DATA IS DENOTED BY THE BLUE CURVE, RUN WITH CONVECTIVE SUPPRESSION OF ONLY DEEP CONVECTION IN CLEAR AREAS USING GOES DATA IS DENOTED BY THE PURPLE CURVE, AND WITH NO GOES DATA IS THE GREEN CURVE. SEGMENT NUMBER 27 IS THE TIME 0200 UTC 5 FEB 2008 WHICH IS THE TIME OF THE HORIZONTAL PLOTS IN FIGURE 6.3. 90
FIGURE 6.2 MODEL CLOUD AGREEMENT INDEX FOR THE 72-H 30-KM WRF SIMULATION FOR THE PERIOD 00 UTC 4 FEBRUARY THROUGH 00 UTC 7 FEBRUARY 2008. RUN WITH CONVECTIVE SUPPRESSION OF DEEP AND SHALLOW CONVECTION IN CLEAR AREAS USING GOES DATA IS DENOTED BY THE BLUE CURVE, RUN WITH CONVECTIVE SUPPRESSION OF ONLY DEEP CONVECTION IN CLEAR AREAS USING GOES DATA IS DENOTED BY THE PURPLE CURVE, AND WITH NO GOES DATA IS THE GREEN CURVE. SEGMENT NUMBER 27 IS THE TIME 0200 UTC 5 FEB 2008 WHICH IS THE TIME OF THE HORIZONTAL PLOTS IN FIGURE 6.3. 91
FIGURE 6.3 PLOTS FOR 0200 UTC 5 FEB 2008. COLOR CODE FOR (A) AND (B): RED, MODEL AND SATELLITE CLEAR, BLUE, MODEL AND SATELLITE CLOUDY; GREEN, MODEL INDICATED CLOUD WHILE SATELLITE WAS CLEAR; AND ORANGE, MODEL INDICATED CLEAR WHILE SATELLITE WAS CLOUDY (SEE TABLE 6.1). (A) MODEL-SATELLITE COMPARISON FOR CASE OF USING GOES DATA TO CONTROL CONVECTION, (B) MODEL-SATELLITE COMPARISON FOR CONTROL CASE OF NOT USING GOES DATA TO CONTROL CONVECTION, (C) MODEL CLOUD TOP PRESSURE (MB) FOR CASE OF USING GOES DATA TO CONTROL CONVECTION, AND (D) GOES CLOUD COVER PERCENTAGE WITH COLORED VALUES GREATER THAN 1 %. SEE TEXT FOR ADDITIONAL DETAILS. 92
FIGURE 9.1 EXAMPLE OF WRF AND WDIAG GRIDS. THE WRF ARAKAWA C-GRID IS DENOTED BY THE "+", "U", AND "V" GRID POINT LOCATIONS. AN EXAMPLE OF THE FINE GRID OPTION IN WDIAG IS GIVEN BY THE GRID POINT LOCATIONS BY THE BOXES AND IN THIS EXAMPLE HAS A HORIZONTAL GRID SIZE HALF OF THE ORIGINAL WRF HORIZONTAL GRID SIZE. SEE THE TEXT FOR ADDITIONAL DETAILS. 120
FIGURE 10.1 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 122
FIGURE 10.2 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 123
FIGURE 10.3 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 124
FIGURE 10.4 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 125

FIGURE 10.5 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 126
FIGURE 10.6 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 127
FIGURE 10.7 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 128
FIGURE 10.8 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 129
FIGURE 10.9 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 130
FIGURE 10.10 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 132
FIGURE 10.11 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 133
FIGURE 10.12 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 134
FIGURE 10.13 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 135
FIGURE 10.14 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 136
FIGURE 10.15 FLOW CHART FOR THE WDIAG CODE. ROUTINE BOXES ARE COLOR CODED IN THE FOLLOWING WAY: 1) MISCELLANEOUS CALCULATIONS, TURQUOISE; 2) GRID CALCULATIONS, VIOLET; 3) MEMORY ALLOCATION, PEACH; 4) INTERPOLATION, BLUE; AND 5) DIVERGENCE CALCULATIONS, GOLD. 137

List of Tables:

TABLE 3.1. WRF CONFIGURATION FOR THE SIMULATIONS 8
TABLE 3.2. CONTINGENCY TABLE USED FOR EVALUATION. 27
TABLE 4.1. CONTINGENCY TABLE CATEGORIES FOR MODEL-SATELLITE CLOUD COMPARISON. 30
TABLE 5.1 MONTHLY AVERAGE STATISTICS FOR AUGUST 2006. TABLE SHOWS FRACTIONS OF GRID POINTS (PIXELS) SATISFYING THE GIVEN CONTINGENCY. 61
TABLE 5.2. THRESHOLD VERTICAL VELOCITIES BASED ON WRF CONTROL SIMULATION. 62
TABLE 2.1 DESCRIPTION OF GOES DATA USED IN WPS/WRF PACKAGE 74
TABLE 3.1 EXAMPLE OF ADDITIONS TO THE NAMELIST.WPS FILE TO PROCESS GOES DATA. 75
TABLE 3.2 SUMMARY OF WPS CODE CHANGES TO PROCESS GOES DATA. SUBROUTINE NAMES IN ORANGE ARE NEW, WHEREAS SUBROUTINE NAMES IN PURPLE ARE ORIGINAL WPS CODE WHICH WAS MODIFIED. 77
TABLE 3.3 ADDITIONS TO METGRID.TBL FILE TO PROCESS GOES DATA 78
TABLE 3.4 EXAMPLE OF CHANGES TO THE NAMELIST.INPUT FILE TO USE GOES DATA. 79
TABLE 3.5 SUMMARY OF WRF CHANGES TO IMPLEMENT USE OF GOES DATA. SUBROUTINE NAMES IN PURPLE DESIGNATE EXISTING SUBROUTINES WHICH WERE MODIFIED. 80
TABLE 3.6 SUMMARY OF CHANGES TO REGISTRY.EM FILE TO ENABLE GOES DATA PROCESSING. 81
TABLE 5.1 LISTING OF FILES FROM TAR FILE TAR.GOES.CHANGES. 84
TABLE 6.1 CONTINGENCY TABLE CATEGORIES FOR MODEL-SATELLITE CLOUD COMPARISON. SAME COLOR SCHEME IS USED IN THE PLOTS (A) AND (B) IN FIGURE 6.3 87
TABLE 2.1 MAIN SUBROUTINE CALLS FROM SUBROUTINE WDIAG. 98
TABLE 3.1 GRID, INTERPOLATION, AND ANALYSIS OPTIONS IN THE "WDIAG_OPTIONS.INP" FILE. 100
TABLE 3.2 PARAMETERS USED WHEN HZ_INTERP_TYPE_OUT = 'GAUSSIAN' IN THE "WDIAG_OPTIONS.INP" FILE. 101
TABLE 3.3. PARAMETERS FOR ANALYTICAL W TARGET IN THE "WDIAG_OPTIONS.INP" FILE. 102
TABLE 3.4. PARAMETERS FOR SEQUENTIAL OVER-RELAXATION (SOR) CALCULATIONS IN THE "WDIAG_OPTIONS.INP" FILE. 103
TABLE. 3.5 PARAMETERS USED IN SUBROUTINE CHECK_VERTICAL_VELOCITIES TO CONTROL MAXIMUM CHANGE IN COLUMN DIVERGENCES IN THE "WDIAG_OPTIONS.INP" FILE. 103
TABLE 4.1. ARGUMENT LIST FOR MAIN DRIVER OF WDIAG. THE COLUMNS, FROM LEFT TO RIGHT, ARE: 1) ARGUMENT NUMBER N; 2) ARGUMENT NAME;3) DIMENSION (0D, 1D, 2D, OR 3D); 4) IO – REPRESENTING IN/OUT; 5) DESCRIPTION; 6) HGRID – HORIZONTAL GRID TYPE (RELATIVE TO ARAKAWA C-GRID, CRS FOR SCALAR-GRID, U FOR U-GRID, AND V FOR V-GRID; 7) VGRID – VERTICAL GRID TYPE – EITHER HALF OR FULL, AND 8) UNITS/COMMENTS. IN COLUMN 8, IF THE UNITS ARE DIFFERENT FROM WRF UNITS OR EXPECTED UNITS THE SCALING FACTOR IS IN SQUARE BRACKETS. 106

Report Type: Final Report
Project Number: **CM5077**
Project Title: Cloud Assimilation into the Weather Research and Forecast (WRF) Model
Author: Arastoo Pour Biazar
Mailing Address: Earth System Science Center, University of Alabama in Huntsville, Huntsville, Alabama 35899
Phone: (256) 961-7970
Fax: (256) 961-7755
E-mail: biazar@nsstc.uah.edu

Report Date: February 28, 2011

1 SUMMARY

The University of Alabama in Huntsville (UAH) was awarded a contract by TCEQ through Lamar University to 1) evaluate the performance of Weather Research and Forecast (WRF) Model with respect to the simulation of clouds as it pertains to air quality applications and establish a baseline model cloud performance; 2) to streamline, transition, and test a satellite assimilation technique for clouds that was previously implemented in MM5 by UAH to WRF modeling system (MM5 is the Fifth Generation Pennsylvania State University/National Center for Atmospheric Research mesoscale model); 3) and to devise and recommend a revised or alternate technique that is easy to implement, computationally inexpensive, and operationally feasible to be used in regulatory air quality applications. This report documents the activities undertaken by UAH to address the objectives of the project and suggests a way forward for future activities.

A baseline WRF simulation for the month of August 2006 was established and was used as the basis for model performance evaluation. The baseline simulation with 36-km grid spacing over continental United States comprised three different convective parameterization schemes, namely Kain-Fritsch (KF), new Grell (G), and Grell-Devenyi (GD). Also based on a sensitivity study using WSM6 and Lin microphysics options, it was decided to use Lin microphysics as it was more suitable in this case study. GOES observations were used to evaluate the baseline model performance with respect to the simulation of clouds and the results indicated that KF performed better than others while G and GD schemes did not show a significant difference. With respect to the standard meteorological variables (i.e., wind, temperature, and moisture) there were no significant differences in model performance between the three convective parameterizations.

In transitioning the satellite cloud assimilation technique to WRF it was found that WRF model does not respond to the adjustments in the same way that MM5 was responding. The technique as implemented in MM5 consists of two major components. The first component deals with creating clouds in the model where the model is under-predicting clouds, and the second component attempts to remove clouds where the model is over-predicting clouds. In the previous MM5 activity, removing clouds from areas of over-prediction was mainly achieved through suppressing convective initiation. As the first step in this project we attempted to suppress convective initiation to clear erroneous clouds. However, in WRF as the subgrid

convective parameterization was suppressed, model created grid scale resolved clouds in response. This caused a reduction in precipitation, but did not correct the over-prediction. As a matter of fact, convective suppression marginally increased cloudiness in the affected regions. This indicated that without adjusting the vertical velocity (to create subsidence in the clear areas), convective suppression alone is not an option in WRF.

Our fundamental approach for correcting cloud fields in the model relies on use of GOES observations of clouds. Therefore, satellite observations are used to both to evaluate the model and identify the locations of over- and under-prediction, as well as determining the key variables (such as vertical velocity) that are needed to be adjusted in the model. First, based on a control (baseline) simulation, using multiple linear regressions, statistical relationships between key model variables (e.g., maximum vertical velocity and its elevation and depth of the layer, cloud depth, cloud top temperature and albedo, etc.) is developed. Second, based on satellite observations of cloud top pressure and cloud albedo, key variables for areas of disagreement between model and observation are calculated. Third, having the target vertical velocity and using a one-dimensional variation technique wind fields are adjusted and used as a nudging field. This approach should be able to correct the cloud fields in the model in a sustainable manner as it adjusts the dynamics and creates an environment conducive to cloud formation or clear sky.

Since one of the objectives of this project was to streamline our technique and reduce the complexity of implementation as well as reducing the computational burden, we proposed to continue on a two track approach for cloud correction. The first path was to continue to modify the previous software and develop new tools to transition the technique as was implemented in MM5 to WRF. And the second path was to implement an alternate simple approach that was devised by revisiting the fundamental problem.

In revisiting our fundamental approach our composite analyses showed that we do not appear to have strong functional relations between simple model maximum vertical velocity in a column and cloud albedo. This perhaps had contributed to the complexity of the implementation of the technique in MM5 and the need for auxiliary adjustments to create or remove clouds. However, it was also found that the original hypothesis is supported by the model. That is, cloudy areas in the model are associated with lifting and the clear areas are associated with subsidence. Based on these analyses it was decided to use model statistics to identify target vertical velocity and moisture needed for create or clear erroneous clouds. This approach constructs a tabular threshold values for maximum vertical velocity and moisture for cloudy and clear areas based on model statistics. Then, based on GOES observations, target vertical velocities are identified for the grid cells that are in disagreement with observations. Based on these target vertical velocities, one-dimensional variation technique will be applied to create a new nudging wind field.

The alternate approach still partially uses some of the main components of the first approach, i.e., tools for statistical analysis and one-dimensional variation technique for adjusting wind field, but avoids the complexity of the first approach and is computationally efficient. For example, in the alternate approach there is no need for adjusting many auxiliary variables. The new approach also eliminates the need for a secondary nudging mechanism in WRF. It also

eliminates the need for cluster analysis that is the most computationally expensive component of the first approach.

Preliminary results from the alternate approach shows improvements in cloud simulations regardless of the convective parameterization used for the simulation. While the preliminary results are promising, the alternate approach still needs to be refined. The technique has indicated that adjusting vertical velocity and moisture has the potential of correcting the cloud fields in the model. There are many issues related to the magnitude and height of target vertical velocity that still need to be addressed. In the work presented here there has been no moisture adjustment and this task will be taken up in the future activities.

Our overall approach for achieving the objectives of this project has been to incrementally improve model cloud simulation by acknowledging and prioritizing the remaining issues to be addressed. A current working alternative for defining target vertical velocity and moisture is to analytically arrive at these targets. This will eliminate the need for relying on model average statistics and will produce more accurate targets for adjustment.

A detailed description of each activity is presented in the following sections. To help readability of the report, more detailed technical information are provided in the appendices. All the tools and software developed under these activities will be packaged and transferred to TCEQ.

2 INTRODUCTION

Clouds play a critical role in the production and destruction of pollutants. However, numerical meteorological models used in the creation of the physical atmosphere in the SIP modeling process have traditionally had significant problems in creating clouds in the right place and time compared to observed clouds. This is especially the case during air pollution episodes when synoptic-scale forcing is weak (e.g. Stensrud and Fritsch 1994).

While the previous activities supported by TCEQ have resulted in improving the radiative effect of clouds in air quality simulations (Biazar et al., 2007), physical inconsistencies remain a concern as the insolation and photolysis fields derived from satellite data do not agree with the model clouds. The purpose of the current activity is to improve model location and timing of clouds in the Weather Research and Forecast (WRF) meteorological model selected for driving photochemical models in future State Implementation Plans (SIPs). This activity provides techniques, using satellite data, to first quantify errors in model clouds, and then to improve the spatial location and timing of clouds in WRF while keeping all other meteorological variables in balance.

The basic approach is to use a GOES cloud image to determine the cloud truth. The strategy will be first to examine differences between model clouds and the GOES clouds. Then two paths will be followed to adjust the model to make the model clouds agree better with the GOES clouds. The first path involves suppressing the calls to the model subgrid cumulus parameterization schemes where the GOES image indicates there are no clouds. The second path is to adjust key model variables such as vertical velocity and relative humidity so that the model would produce clouds where the GOES image has clouds and to suppress model clouds where the GOES image says it is clear.

The first step in this project was to establish a baseline simulation that will be used for further model evaluations. The baseline simulation will be used as the reference for evaluating model performance in subsequent simulations where satellite observations are used to adjust model clouds.

In the following chapters, first the baseline simulation is documented, and then the technique and basic metrics used for performance evaluation will be described. Subsequently, activities in porting the technique from MM5 to WRF will be discussed, and then the new alternate approach will be presented.

3 BASELINE SIMULATION

3.1 Model Configuration

WRF simulations in this project span over August 2006. This period coincides with TexAQS-II field study, has been used for the previous modeling studies, and offers a substantial observational dataset for model evaluation. In addition to observations, model performance was evaluated against previous MM5 simulations on a similar grid structure and comparable configuration to check for the consistency of predictions. In order to select a baseline simulation that yields the best performance with respect to cloud simulation, three different set of simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization were performed to investigate the impact of these schemes on the overall cloud prediction.

WRF (version 3.1.1) was used for simulations over the continental United States (CONUS) for the period of June 6 through September 24, 2006. The coarse domain has a spatial grid spacing of 36 km x 36 km horizontally (164 grids in west-east direction and 128 grids in south-north direction) and a non-uniform vertical structure with 42 levels (41 layers) with the top pressure at 50 mb. The non-uniform vertical structure is designed to have high resolution within the boundary layer (and close to the surface) and near tropopause (8 to 14 km) in order to better explain the stratospheric-tropospheric exchanges.

NCEP Eta Data Assimilation System (EDAS) analyses were used in WPS to create the initial and boundary conditions for the simulations. Table 3.1 summarizes the key options used for the simulations. To test the sensitivity of the results to the choice of microphysics option, an additional simulation using the WRF Single-Moment 6-Class (WSM6) microphysics scheme (with KF option only) was also performed. WRF FDDA (four dimensional data assimilation) was also utilized to nudge the model toward analyses data for better performance in the baseline simulation. We strived to have the configuration of the baseline simulations similar to the common practices in air quality modeling for the State Implementation Plan (SIP) in Texas. Simulations were performed in 5.5 day segments and re-initialized from analyses for each segment at 0 GMT. The first 12 hr of each segment is discarded as the spin-up time and the rest of the output is appended to the previous segments to create a continuous record.

Table 3.1. WRF configuration for the simulations

	Domain 01	Domain 02
Running period	June 6th - September 24th	
Horizontal resolution	36 kn	12 km
Time step	90s	30s
Number of vertical levels	42	
Top pressure of the model	50 mb	
Shortwave radiation	Duhia	
Longwave radiation	RRTM	
Surface layer	Monin-Obukhov similarity	
Land surface layer	Noah (4-soil layer)	
PBL	YSU	
Microphysics	LIN, WSM6	
Cumulus physics	Kain-Fritsch/Grell-Devenyi/new Grell	
Grid physics	Horizontal wind	
Meteotological input data	EDAS	
Analysis Nudging	yes	
U, V Nudging Coefficient	3x10-4	
T Nudging Coefficient	3x10-4	
Q Nudging Coefficient	10-5	
Nudging within PBL	Yes for U and V, NO for q and T	

3.2 Establishing the Baseline Simulation

The results were evaluated in several ways. First the model results with respect to temperature, relative humidity, winds, and pressure surfaces were compared with the previous MM5 simulations for this period. MM5 simulations for this period have been used in previous studies funded by HARC/TCEQ and extensively evaluated. Therefore, a consistency between WRF and MM5 will indicate that WRF has been able to reasonably capture the synoptic scale features. Weather charts were also used to compare the observed synoptic patterns with model predictions. The second set of evaluations evaluates the model results in more detail using the surface observations. METSTAT was used to create the statistics for key meteorological parameters and also to examine the time series for selected locations. The third set of evaluations, involve the use of geostationary satellite observations of clouds to examine model performance with respect to cloud prediction.

3.3 Comparison with MM5 and Weather Charts

The comparison of the current WRF simulations with the previous MM5 Simulations showed general agreement between the two simulations. Most of the discrepancies seem to be due to a discrepancy in processed analyses fields used for the two simulations. While the same analyses fields are being used for the two simulations, the preprocessors and the options used in pre-processing could have been partially responsible for the discrepancy. Here we present few snap shots of the results as examples. These examples are from the simulation that used KF

convective parameterization. The performance of the simulations using other convective parameterizations on the synoptic scale was similar.

To demonstrate the similarities between MM5 and WRF simulations and their reasonable performance on synoptic scale the example covers the period of August 19 through August 21 where a stationary front extending from southwest to northeast Continental U.S. was present. MM5 results for this period have been extensively evaluated in a previous HARC project (http://files.harc.edu/Projects/AirQuality/Projects/H109/H109FinalReport.pdf). Thus, we expect WRF to perform as well as MM5 for this period.

Models generally have difficulties during the periods where the synoptic forcing is weak and this period in August presents such a scenario. **Figure** Error! Reference source not found. shows surface analysis for 21 GMT, August 19, 2006, and the corresponding results from WRF and MM5 simulations. The plots show the predicted temperature and winds as well as corrected surface pressures from the two model simulations. As it can be seen, the general weather patterns in the two simulations and the temperatures are in agreement. They are also in general agreement with the surface observation chart presented in the figure.

The stationary front was over northeast Texas during August 21-22. Again both WRF and MM5 show a reasonable agreement with surface weather charts for this period. This is shown in **Figure 3.2** where the weather chart for 21 GMT, August 20, 2006, together with MM5 and WRF results are demonstrated. There are small discrepancies in wind fields between the two models. These discrepancies are partly due to the differences in the boundary condition and analysis nudging in the two simulations. There are also differences in temperature and other fields in some areas, but the overall synoptic pattern of the weather systems in the two simulations is similar to that of the observation.

Comparing the results from the two models for other periods in August also indicated good agreement in temperature and wind fields and the differences were negligible.

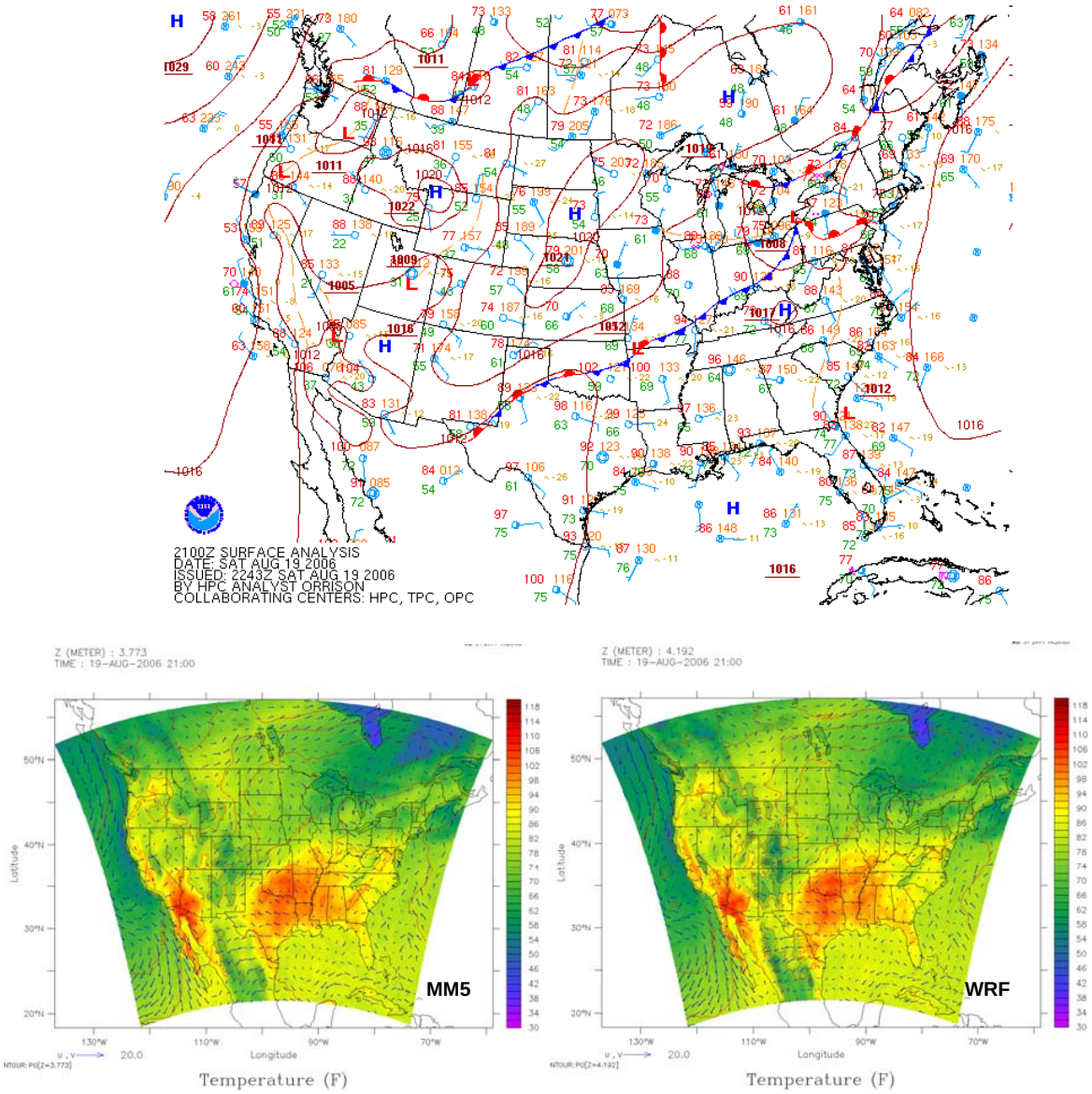


Figure 3.1. Surface analysis (top), MM5 simulation (lower left), and WRF simulation (lower right) results for 21 GMT, August 19, 2006 where a stationary front extending from southwest to northeast continental U.S. was present.

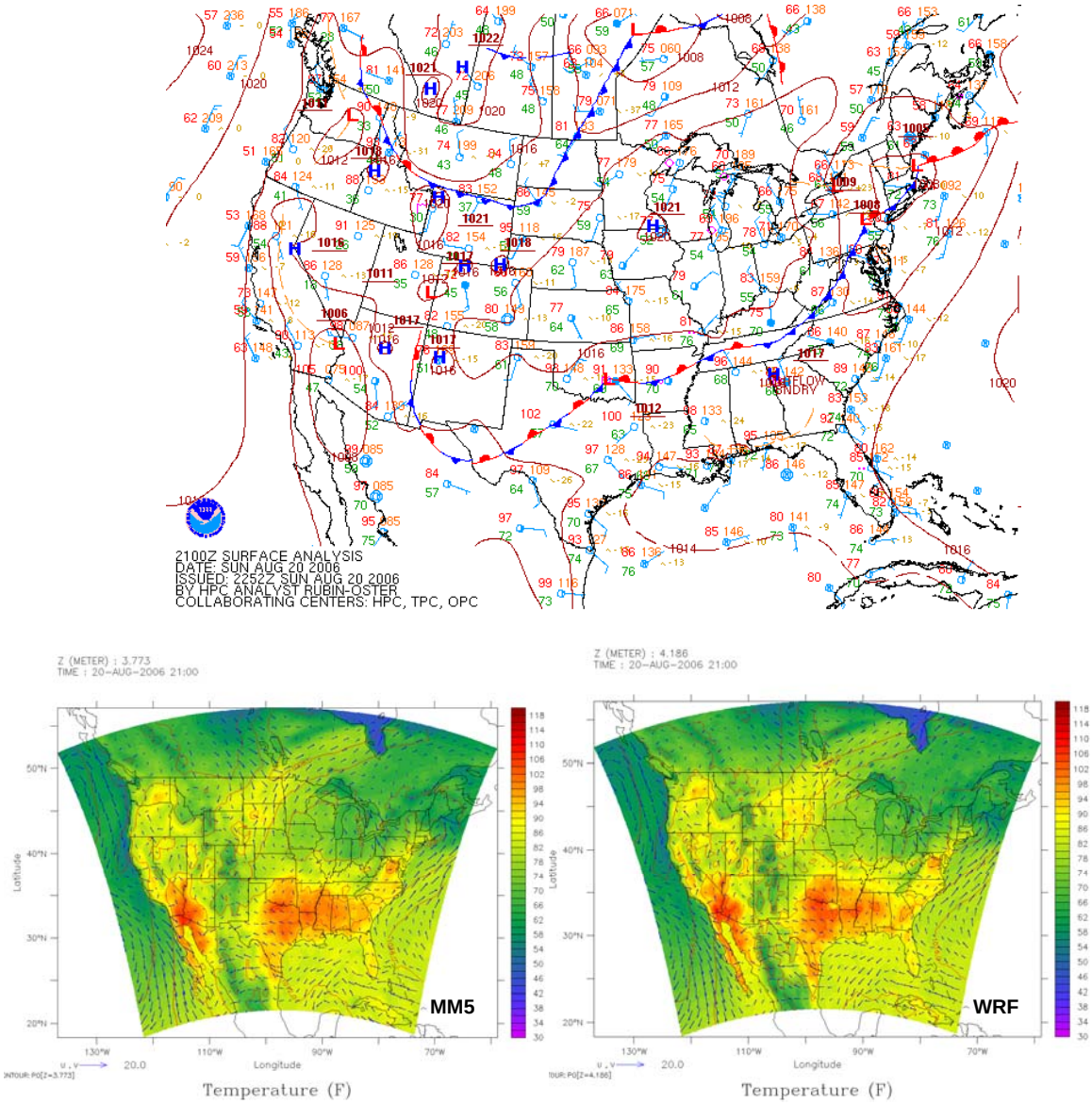


Figure 3.2. Surface analysis (top), MM5 simulation (lower left), and WRF simulation (lower right) results for 20 GMT, August 20, 2006.

3.4 METSTAT Evaluation

The second set of evaluations was performed using METSTAT software from Environ (http://www.camx.com/files/metstat.27oct09.tar.gz). These evaluations attempted to quantify the impact of the three different convective parameterizations, KF, GD, and NG, on the prediction of standard atmospheric variables. The overall conclusion from these evaluations is that the performance of all three parameterizations was comparable with respect to the domain-wide statistics of temperature, relative humidity, wind speed and direction. The additional simulation which used WSM6 microphysics option was also evaluated using METSTAT. National Weather Service (NWS) surface observations were extracted and used in METSTAT.

For better presentation and for figures to be legible, domain-wide statistics presented here covers the first two weeks of August 2006. The results from the second half of August are similar to the first half. **Figure 3.3** shows the domain-wide statistics for temperature for the first two weeks of August 2006. All three convective parameterization (CP) schemes under predicted the temperature, with KF having the lowest negative bias while the RMSE not showing a considerable difference. Both GD and NG perform relatively the same as far as temperature prediction is concerned. With respect to mixing ratio, the results are reversed with GD and NG having a lower bias than KF while they all under predict mixing ratio (as indicated in **Figure 3.4**). The average temperature bias for all three schemes is less than .3 °C while the average mixing ratio bias is less than .6 g/kg.

All three schemes performed similarly with respect to wind speed, with no considerable difference in bias. The results are depicted in **Figure 3.5**. The simulations show over-prediction with an average of less than .5 m/s. The biases in temperature, mixing ratio, and wind speed do not exhibit any correlation and therefore are not caused by any particular weather system. However, the time series analysis of stations over Texas during August 19-22, when a stationary front was present over north Texas, exhibits some spatial inhomogeneity in the model performance.

To evaluate the results during the stationary front episode, time series of the temperature, mixing ratio, and wind speed from all three WRF simulations over Texas were examined. The locations of the stations over Texas are marked on the map in **Figure 3.6**. **Figure 3.7** shows the time series of the temperature, mixing ratio, and wind speed from all three WRF simulations over Texas during the stationary front period. Model results corresponding to the location of the surface stations that had valid measurement were extracted and averaged. The figure also presents the average observations (open circles) for this period. There is no significant difference between the temperature and wind speed predictions for different model configurations and on most days model temperatures shows good agreement with the observations. While model is able to explain the diurnal variation of wind speed reasonably, the best agreement with the observations is on August 19, 21, and 22. With respect to mixing ratio, model underestimates mixing ratio for August 18, 20, 21, and 22, while on August 19 it over-estimates mixing ratio. KF scheme shows the largest under-estimation of mixing ratio. **Figure 3.8** shows an example of model performance over a station in the central Texas (KBAZ) where model shows a reasonable agreement in temperature, mixing ratio, and wind speed. In general model performed better for central Texas than the areas to the west/northwest Texas.

Examining the time series of model predictions from the stations to the north of the stationary front, e.g., KAMA, to the stations in the vicinity of the front, e.g., KAFW and KADS, to the south and away from the front, e.g., K11R, K3T5, and KBAZ, indicates that model performance improves as we move to the south and away from the front. The wind speed results for these stations are presented in **Figure 3.9** and **Figure** 3.10. These figures show the results from simulations using Kain-Fritsch convective parameterization together with Lin or WSM6 microphysics options. As evident in the figures, model doesn't exhibit a noticeable sensitivity to the choice of microphysics option. While there is large variability in observed wind speeds, the

model exhibits a general agreement in diurnal variation and in the magnitude of wind speeds for the stations in the south.

Model greatly under predicts wind speeds at KAMA which is to the northwest of the front. It seems that the model has a broader area of stagnation with weak convergence along the stationary front, while the observation indicates higher winds with some rotation to the north of the front (as evident in **Figure 3.2**). At KAFW and KKADS, the stations that are in the vicinity of the front, model performs better on the average but still is unable to explain the observed variability. The performance is better prior to August 21. On August 21, when the stationary front is over these stations, model under predicts winds.

Figure 3.10 shows the results for K11R, K3T5, KBAZ. These stations are in south Texas and were away from the stationary front (to the south of the front) during this period. Model performs reasonably well for these stations and is able to explain the diurnal variation.

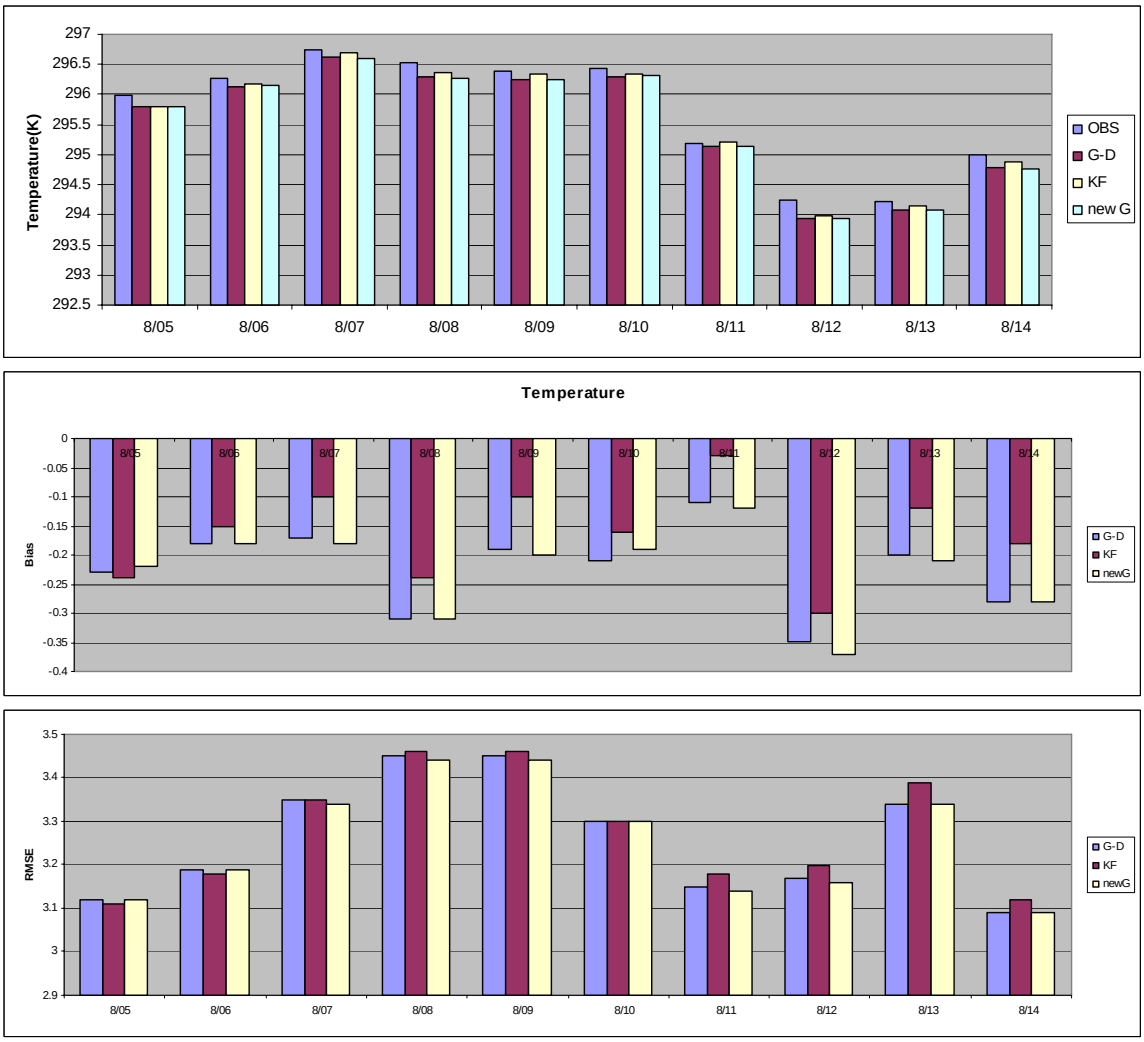


Figure 3.3. Temperature statistics for the first part of August 2006 from WRF simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization.

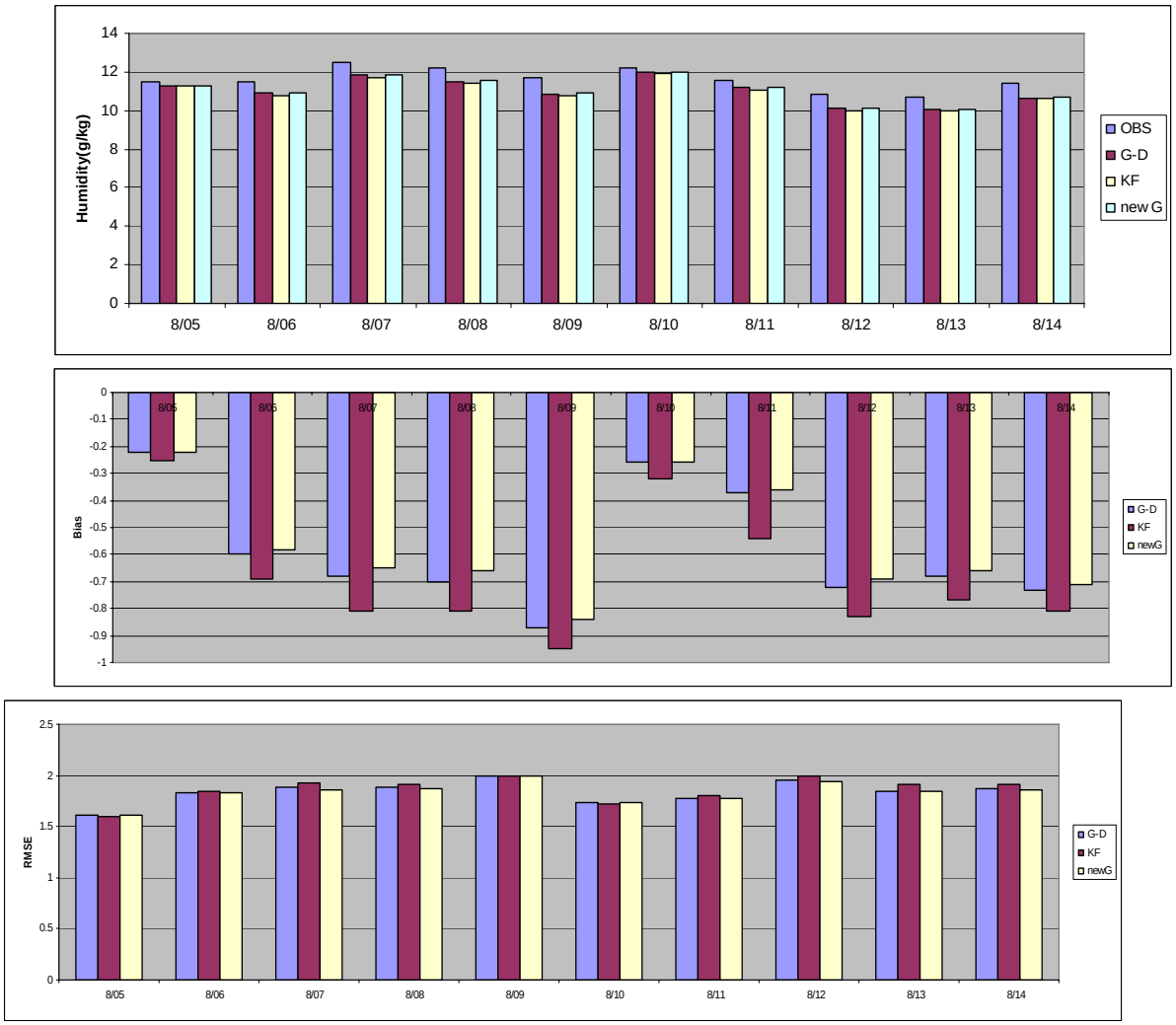


Figure 3.4. Mixing ratio (g/kg) statistics for the first part of August 2006 from WRF simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization.

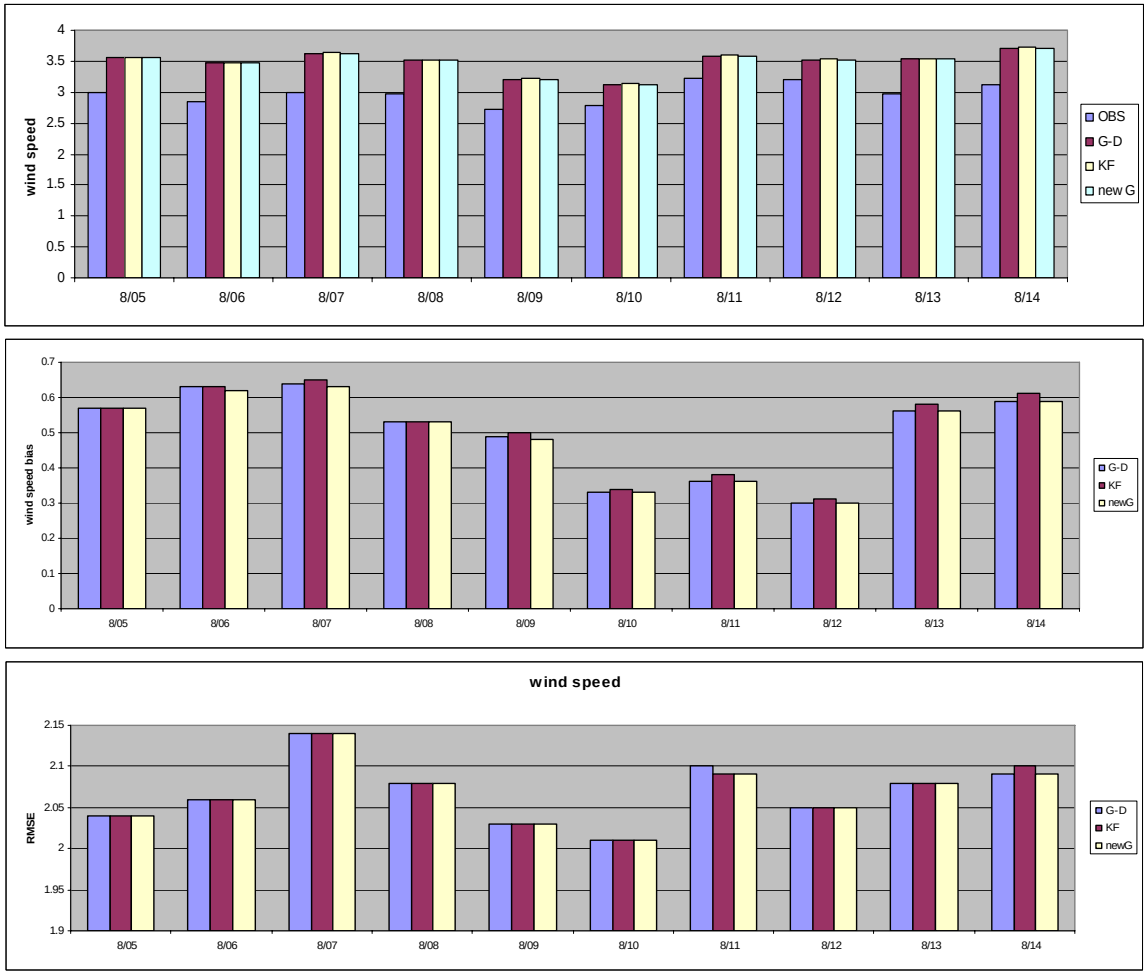


Figure 3.5. Wind speed statistics for the first part of August 2006 from WRF simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization.

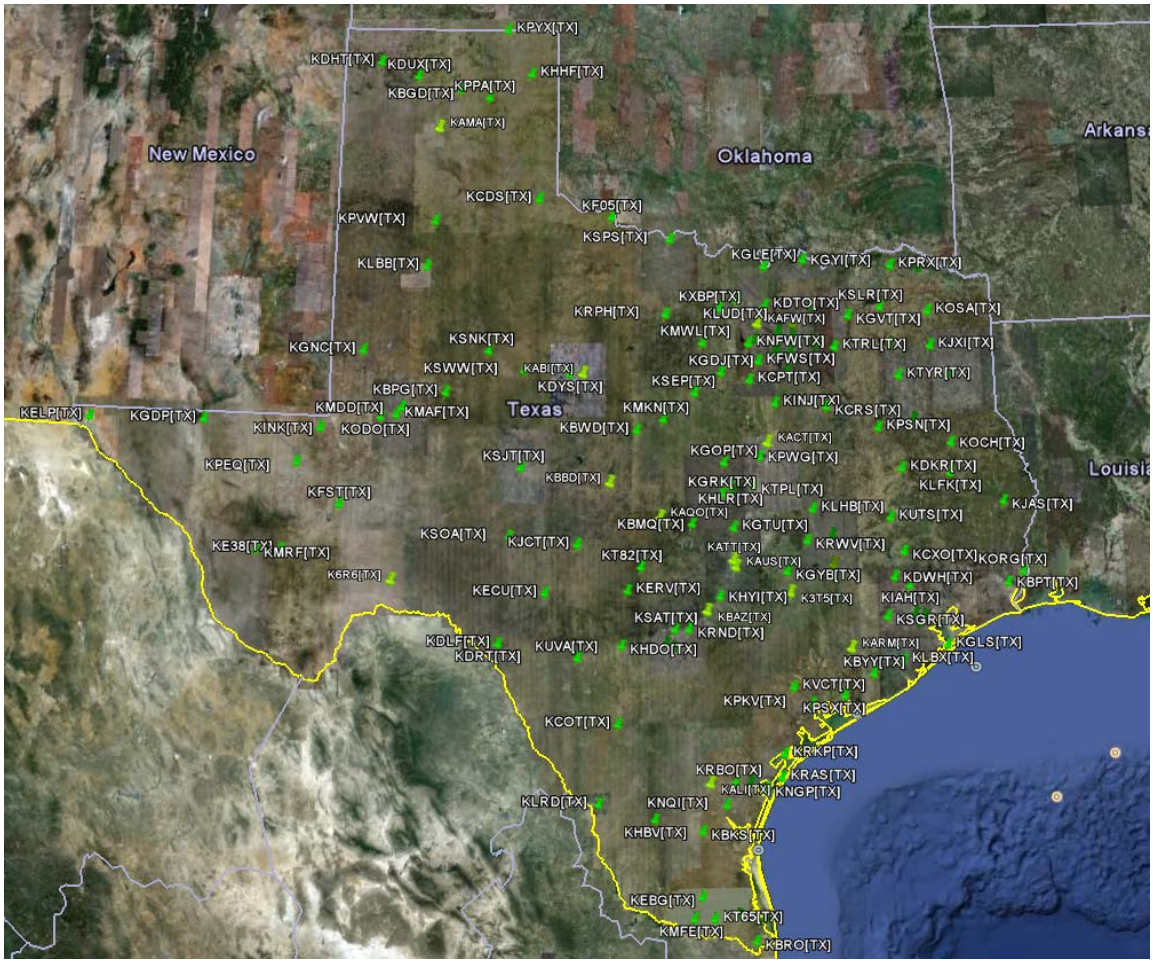


Figure 3.6. Location of weather stations in Texas that was used in the analysis.

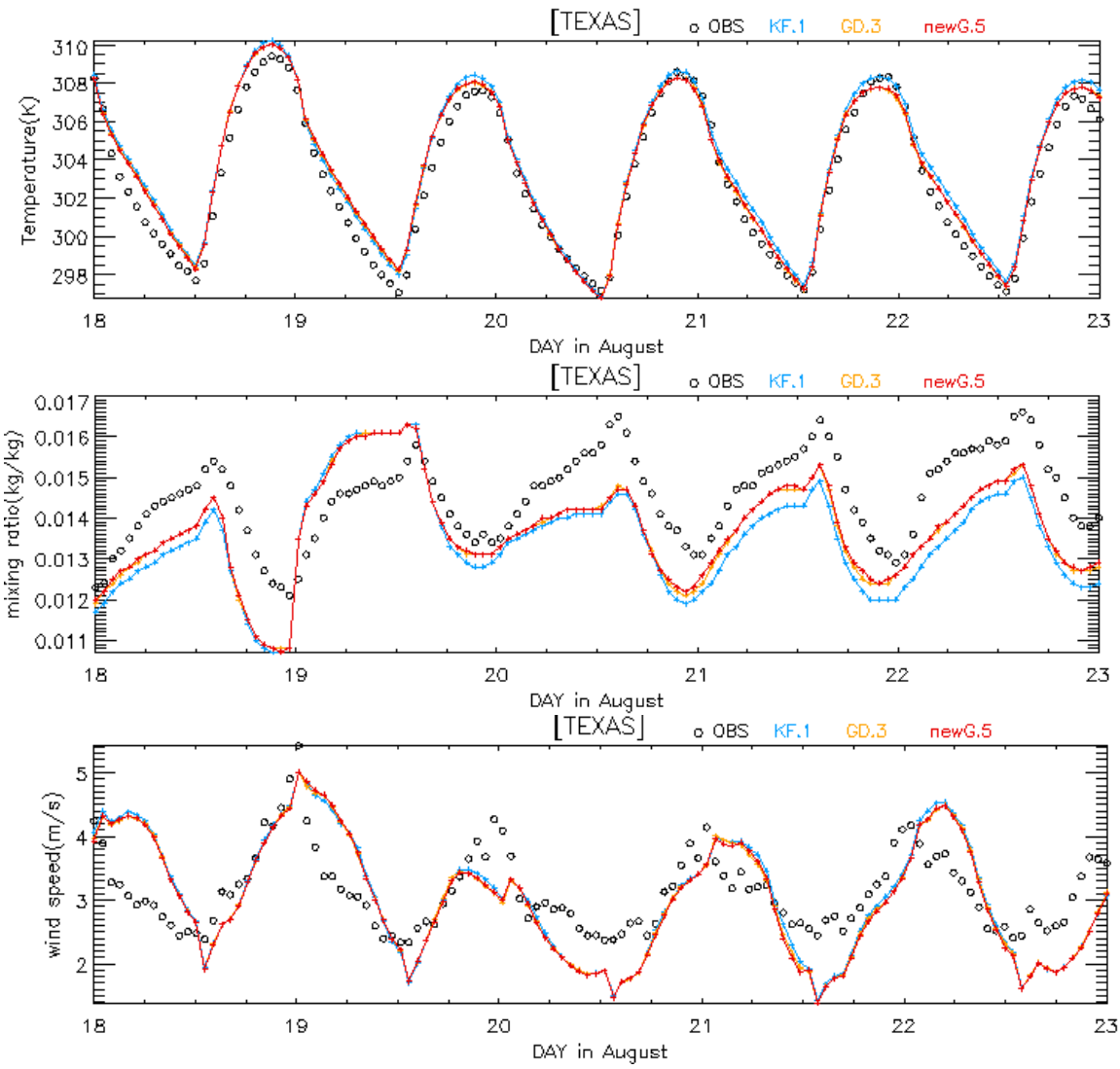


Figure 3.7. Average time series for temperature, mixing ratio, and wind speed during August 18-23, 2006 over stations in Texas. The results from WRF simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization are plotted over surface observations (open circles).

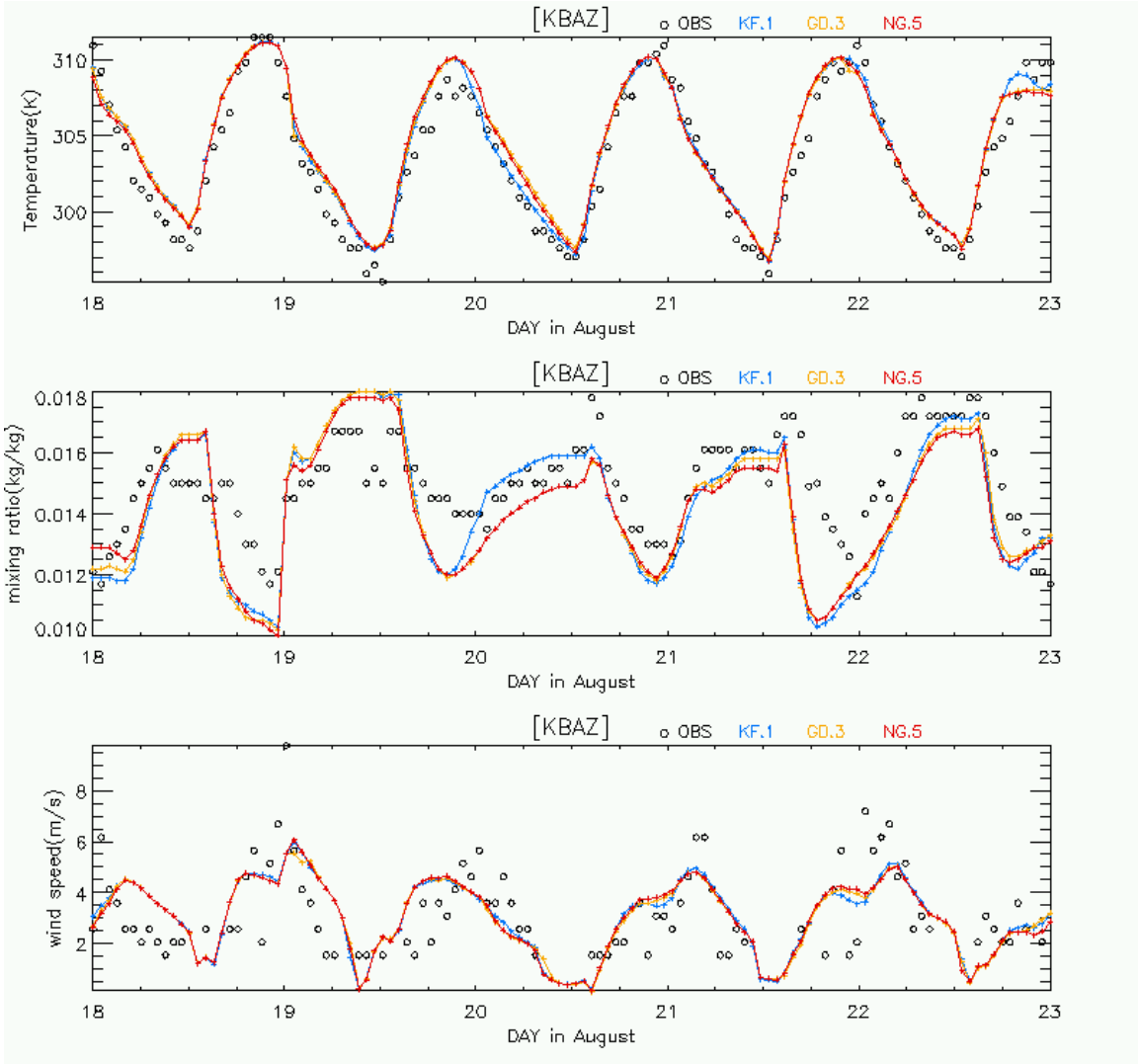


Figure 3.8. Time series for temperature, mixing ratio, and wind speed during August 18-23, 2006 over a station in southwest Texas. The results from WRF simulations using Kain-Fritsch, Grell-Devenyi ensemble, and new Grell scheme for convective parameterization are plotted over surface observations (open circles).

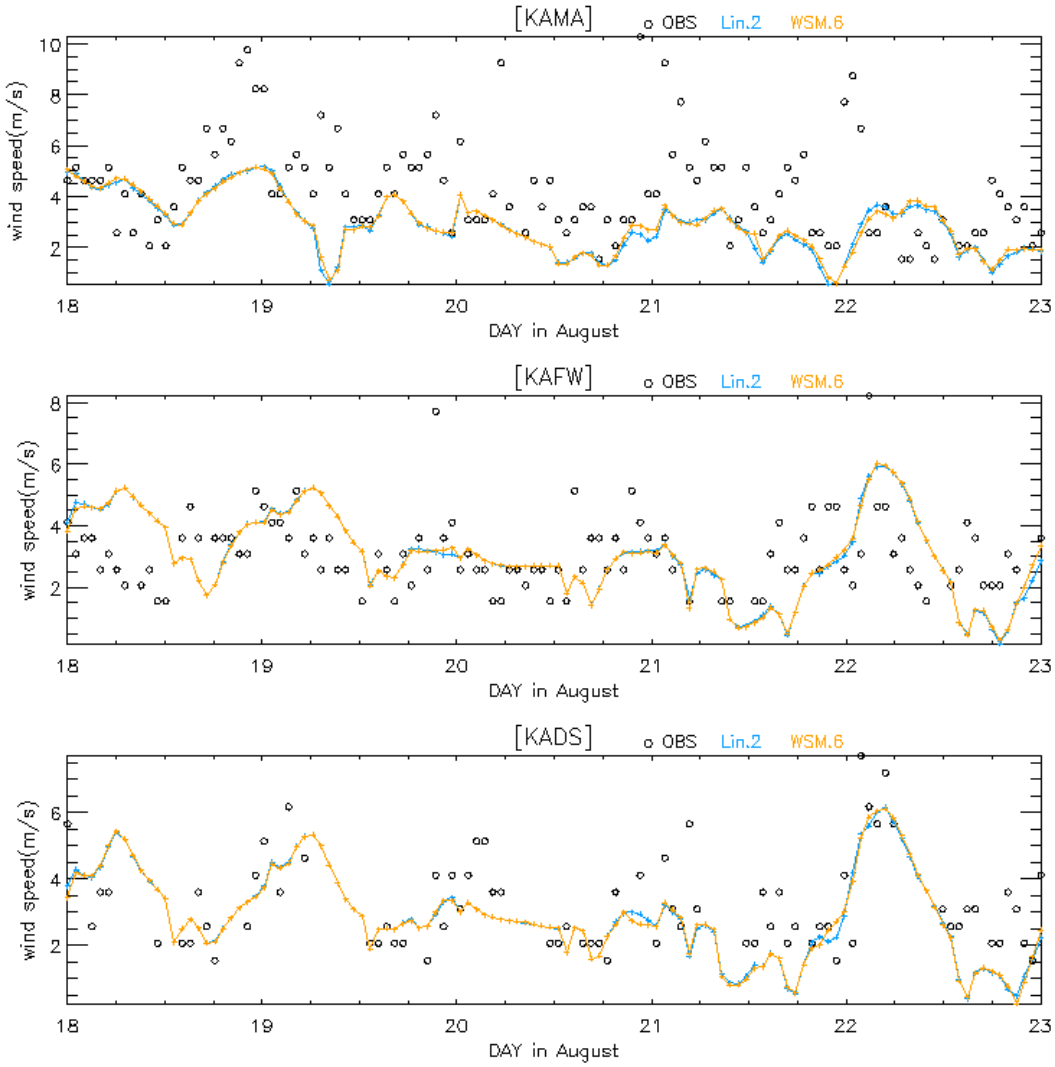


Figure 3.9. Time series for wind speed during August 18-23, 2006 over stations in north Texas. The location of the stations are marked in Figure 7. The results from WRF simulations using Kain-Fritsch convective parameterization together with Lin or WSM6 microphysics options are plotted over the surface observations (open circles).

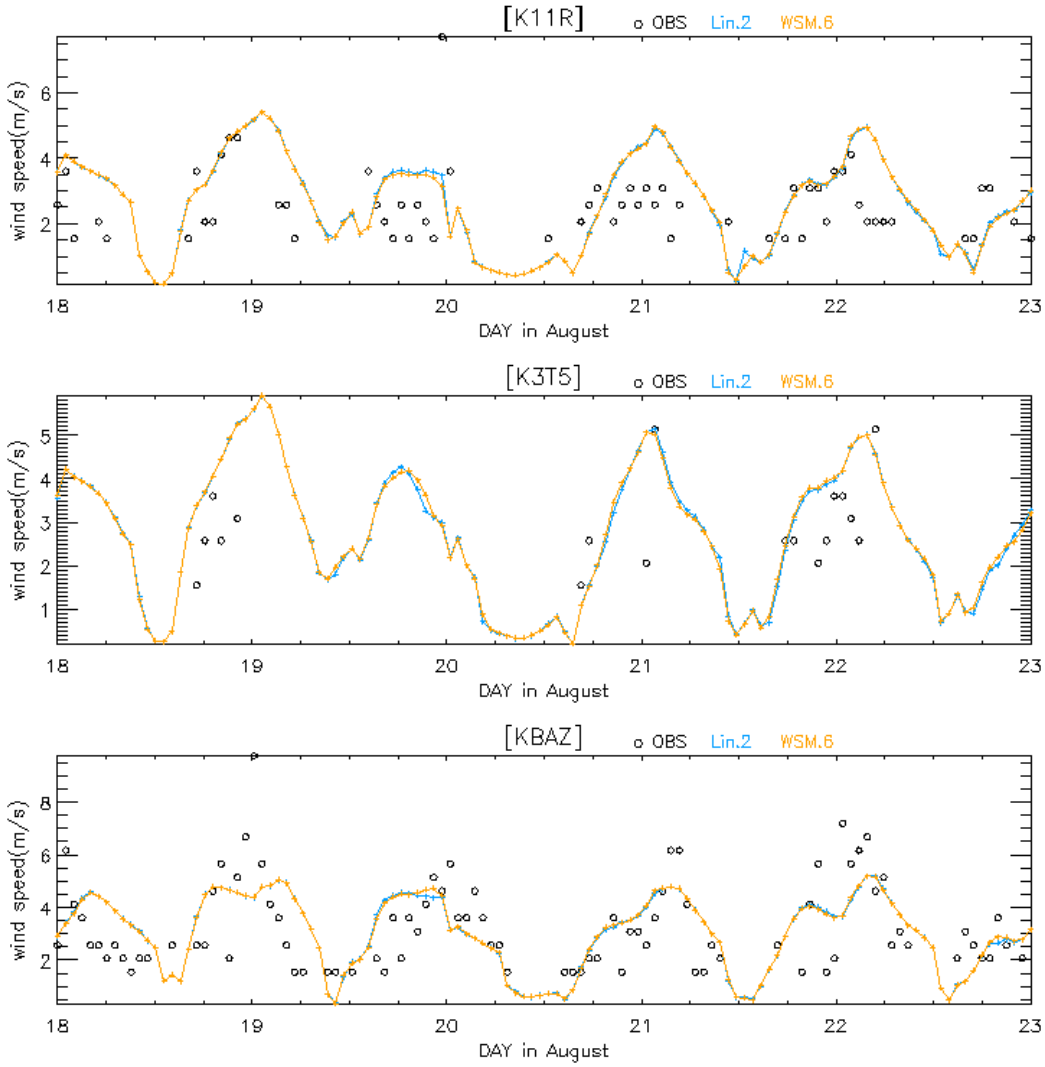


Figure 3.10. Time series for wind speed during August 18-23, 2006 over stations in south Texas. The location of the stations are marked in Figure 7. The results from WRF simulations using Kain-Fritsch convective parameterization together with Lin or WSM6 microphysics options are plotted over the surface observations (open circles).

3.5 Evaluating Cloud Prediction Using GOES Observations

The third set of evaluations was performed against GOES observations of clouds. While surface monitors represent point measurements and are not comparable with the model grid average quantity, satellite observations are aggregated pixel quantity and offer a more comparable measurement. GOES measures the radiative impact of clouds directly in infrared and visible channels. For this evaluation work, derived surface insolations from GOES visible channel were used. A byproduct of surface insolation is cloud albedo that is readily available to be used. However, to create a consistent comparable field between the model and satellite observations, we define an effective cloud index to indicate cloudiness. To arrive at effective cloud albedo, first the clear sky insolation is defined as:

$$R_{\max} = S_0(1-\alpha)$$

Where R_{max} is the clear sky incident shortwave radiation at the surface, S_0 is the solar constant, and α is the surface albedo. Now, the effective cloud index can be defined as:

$$\alpha_c = \left(1. - \frac{R}{R_{\max}}\right)$$

This quantity will also include the small cloud absorption and indeed will yield a normalized index when R_{max} represents the maximum clear sky insolation for any given point and time. To find R_{max} for the model and satellite observations, a composite image that presented the maximum hourly insolation for the month of August was constructed. Then, the cloud index was calculated for each hour (model output time) based on the above formula. The cloud index approaches zero for clear sky condition and increases toward the limit of 1 for more opaque clouds. While the value of the index for the model could be different from that of the observations, it can be a good indicator of cloudiness regardless of the opaqueness of the cloud.

Figure 3.11 presents a snapshot of cloud index for WRF (with KF option) and GOES respectively for 20 GMT on August 16, 2006. The areas of agreement and disagreement are colored in panel c according to the contingency table in the figure. The areas where both model and observations indicate clear sky are marked in blue, while the areas where both model and observations indicate cloudy sky are marked in red. The areas where model is over-predicting clouds are marked in green, while the areas of under-prediction are marked in orange. For this scene WRF predicts that about 38% of the area is cloudy, which is in reasonable agreement with the observations (36% for GOES). However, there is about 30% disagreement with the observations over the spatial distribution of the clouds. Such large disagreements in the spatial distribution are of interest as it can significantly affect the air quality simulations.

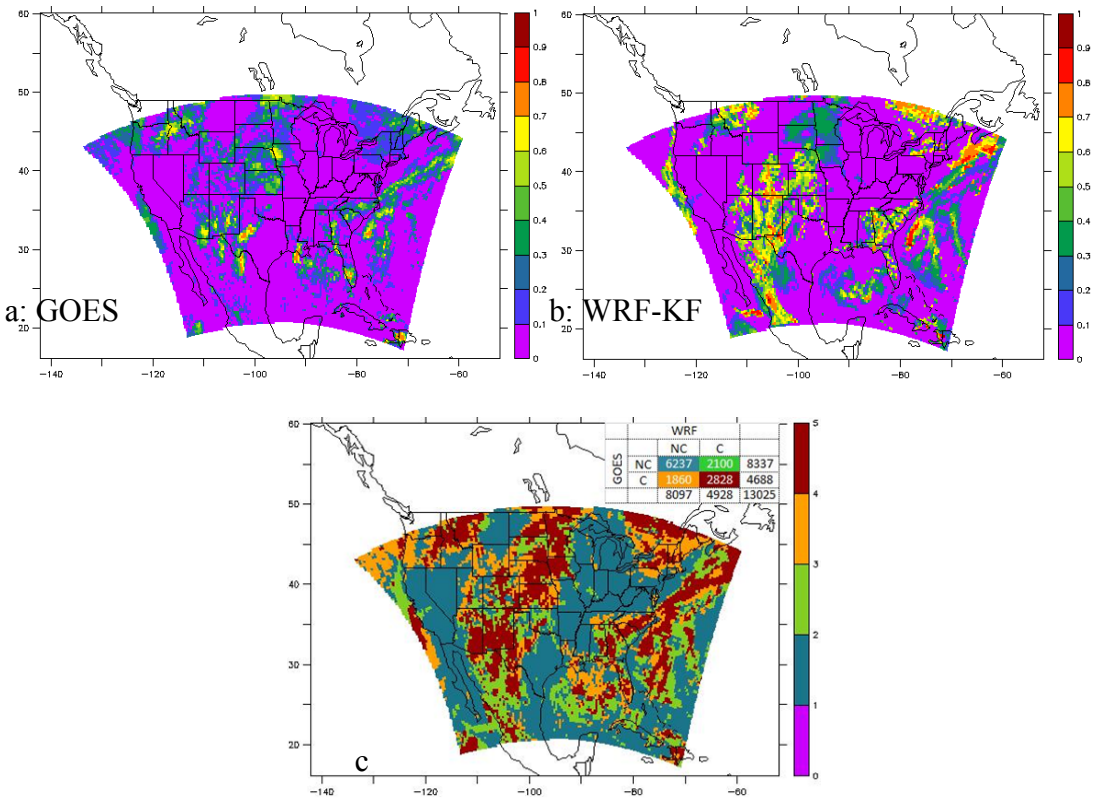


Figure 3.11. Observed satellite cloud versus model cloud for 20 GMT, August 16, 2006. Cloud index for a) GOES observation, b) WRF model simulation using Kain Fritsch convective parameterization, and c) color coded areas of agreement (red and blue) and disagreement (green for over-prediction, and orange for under-prediction) between model and observation are presented in the figure. This case has almost perfect agreement in the fractional cloud/clear area, with WRF showing 38% of the area being cloudy versus 36% for GOES. But there is only about 70% agreement with the observed clouds (spatial disagreement between GOES and WRF is about 30%).

Another example showing the difference between the three convective parameterization options is presented in **Figure 3.12**. The table on each panel in the figure shows the number of grids that are in agreement with observations both for cloudy and clear sky scenes, as well as the number of grids where the model under/over-predicted the clouds. Kain-Fritsch option performs slightly better than New Grell and Grell-Devenyi schemes, as both NG and GD schemes slightly over-predict clouds. But the patterns at this scale are very similar.

The overall model performance is captured in **Figure 3.13**, where the average daily fractional cloud coverage and agreement with the observations are presented. This is the fraction of total model grid cells within GOES window that are covered by cloud. It should also be noted that the daily time window for comparison were chosen as 15-23 GMT so that there is a complete reliable coverage by GOES. The fraction of the domain covered by cloud remains about 30-40% until mid-August with little variation and gradually increases to about 70% by the end of August. This trend is captured by the model with all three convective parameterizations. What is noticeable is that all three convective parameterization schemes over-predict cloud coverage with Kain-Fritsch consistently performing better and predicting a cloud coverage that is closer to the observation. GD and NG options consistently over-predict cloud coverage up to 7% and are

similar in performance with NG performing slightly worse. The model (all three schemes) performs better toward the end of August as the fractional cloud cover increases. The difference in the performance of different schemes also reduces as the cloud cover increases.

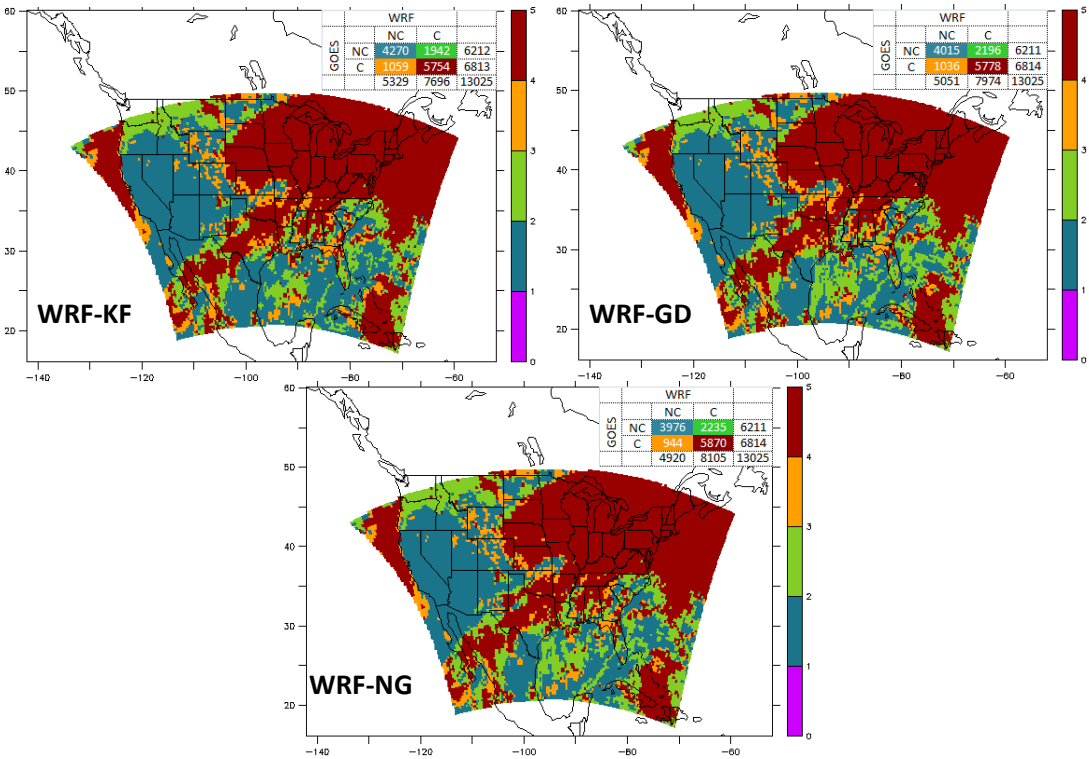


Figure 3.12. Color coded areas of agreement (red and blue) and disagreement (green for over-prediction, and orange for under-prediction) between model and GOES observation for 20 GMT, August 28, 2006. The agreement is 77% for Kain-Fritsch (KF) convective parameterization, versus 75% for Grell-Devenyi (GD) and 76% for New Grell (NG) options. There is no significant difference between GD and NG for this scene. Tabulated values for the number of grids in agreement and disagreement are also presented in the figure.

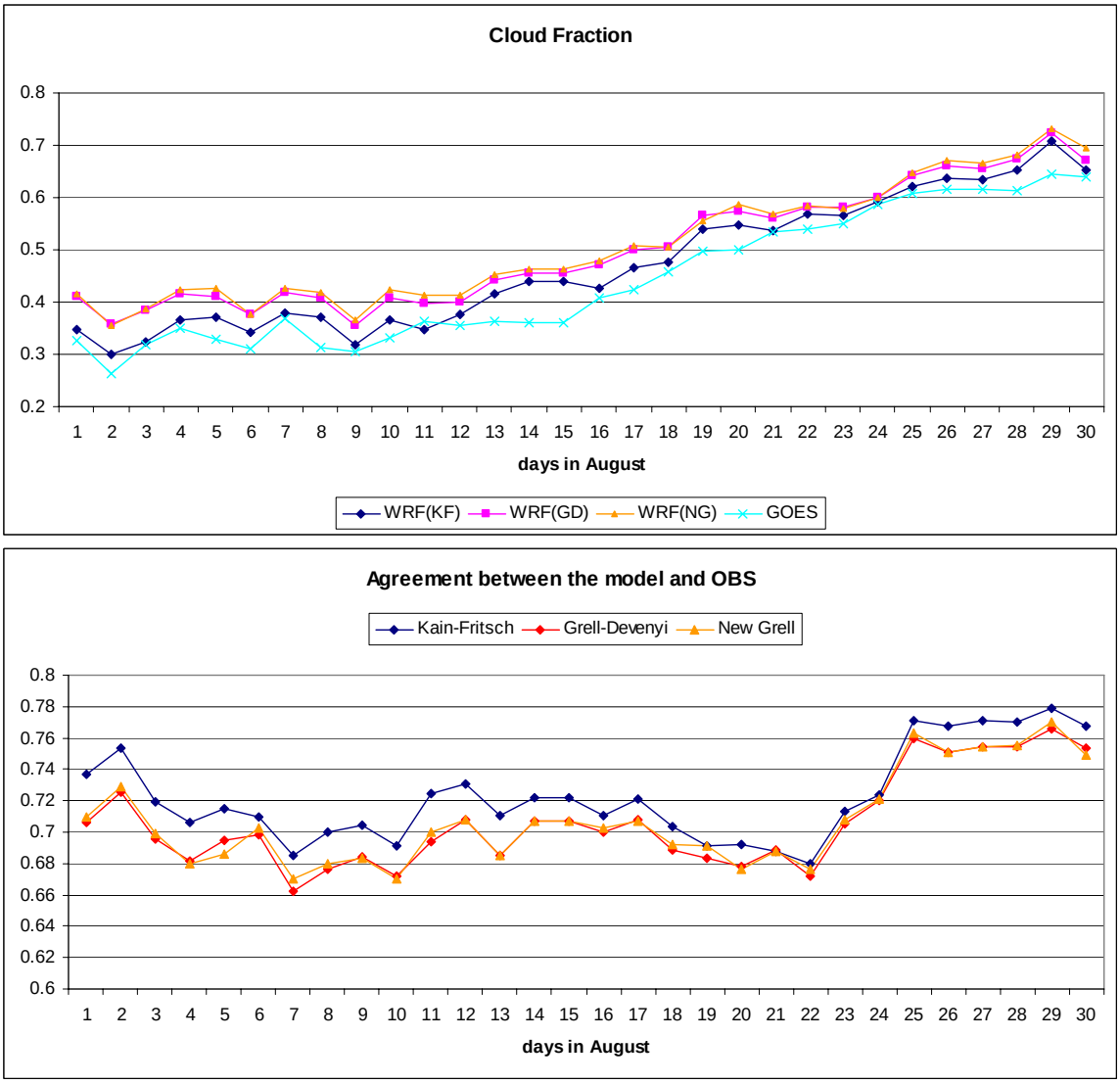


Figure 3.13. Top panel shows the fractional area of the domain that was covered by clouds for each day of August for GOES and each of the WRF simulations. The bottom panel exhibits the daily fractional agreement between the model and GOES observation for each WRF simulation.

The lower panel in **Figure 3.13** shows the fractional agreement between the model and the observations for all three schemes. Again KF exhibits better agreement with GOES observations and GD and NG perform similarly. The agreement for KF starts around 76% in early August, drops to 68% by July 22 and increases to 78% by the end of August. GD and NG options show the same trend, starting with 72% agreement in early August, dropping to 68% by July 22 and increasing to 76% by the end of August. The lowest agreement with observations for all three schemes is on August 22 where the agreement for all three schemes is at 68%. The agreement for all three schemes increases considerably (to 76-78%) after August 22 as the fractional cloud coverage increases. The difference in performance between the three schemes is also reduced as the cloud coverage increase.

To better understand the areas where the model disagrees with the observations, daily statistics of over-prediction and under-prediction is presented in **Figure 3.14**. These are the areas where the model has cloud while GOES observation shows no cloud and vice versa. KF simulation has an over-prediction that stays around 15% and in some areas under-predict clouds by about and average 12%. The performance is relatively consistent with respect to over-/under-prediction of clouds. It also shows small reduction in over-/under-prediction after August 22 where the cloud cover increases. GD and NG options indicate larger gap between over-prediction and under-prediction and perform similarly. The over-prediction is consistent around 20% and under-prediction remains about 12%.

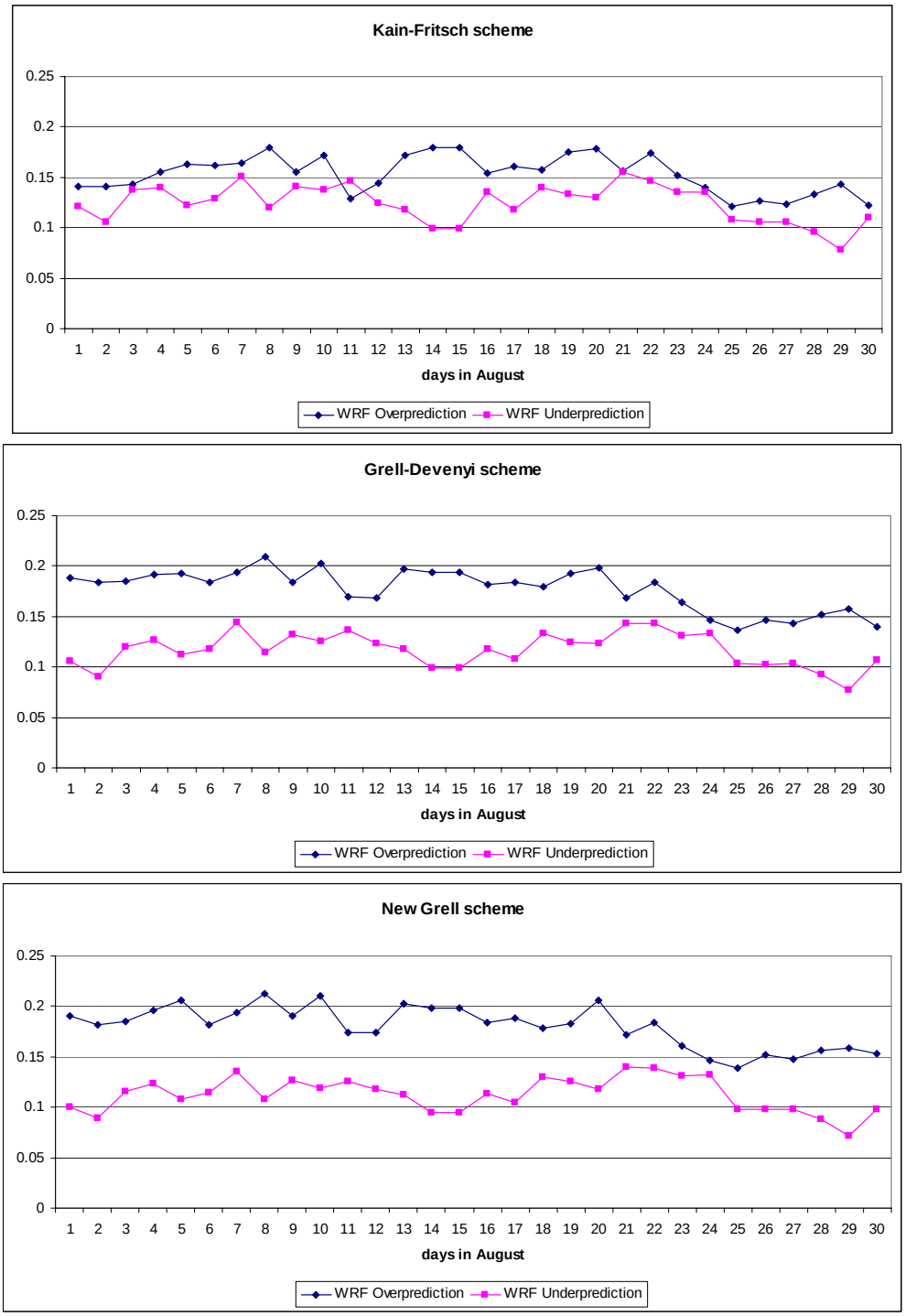


Figure 3.14. WRF over- and under-prediction of clouds presented as the fractional area of coverage for each of the convective parameterization options. In general model over-predicts clouds in more areas than under-predicting clouds. GD and NG show a larger gap between the areas under- and over-predicted than KF option.

3.6 Conclusions from Baseline Simulation

Several WRF simulations for August 2006 were performed in order to establish a baseline simulation for this project. The simulations used a configuration similar to what customarily is practiced in air quality regulatory practices. The simulations differed in convective parameterization and microphysics schemes. A total of 4 simulations were performed with KF/Lin, KF/WSM6, GD/Lin, NG/Lin for convective parameterization/microphysics option.

The overall conclusion is that KF/Lin simulation performed better in predicting cloud cover. There were no significant differences between the simulations with respect to standard meteorological variables evaluated by METSTAT package. It should be noted that these simulations are performed on a relatively coarse grid spacing of 36 km. It remains to be seen if the same conclusion can be reached for a nested simulation with 12 km grid spacing.

There are two noticeable features in the month of August. First is the steady increase in cloudiness from mid-August to the end of the month. Second is the better model performance after August 22 that is preceded by the lowest agreement with observations on August 22. Weather charts for the latter part of August was not examined. Perhaps examining the weather charts can shed light on the kind of weather systems over the domain for this period.

We have also introduced a new metric for model evaluation against GOES observations. This metric is called the Index of Agreement (AI). Table 3.2 shows the contingency table used for this index. The table shows the number of grids where both model and GOES indicate cloudiness (A), number of grids where both model and GOES indicate clear sky (D), number of grids where the model is under-predicting clouds (B), and number of grids where the model is over-predicting clouds (C).

Table 3.2. Contingency table used for evaluation.

		WRF		TOTAL
		Cloudy	Clear	
GOES	Cloudy	A	B	A+B
	Clear	C	D	C+D
TOTAL		A+C	B+D	A+B+C+D

AI is defined as the percentage of grid points that show agreement between model and GOES observations. Based on Table 3.2 AI is defined as:

$$AI = (A+D)/(\text{Total Number of Grids})$$

Where:

A = Number of grid points where both GOES and WRF are cloudy
D = Number of grid points where both GOES and WRF are clear
Total = A+B+C+D = Total number of model grids

The index varies between 0 and 1, where closer to 1 means better performance.

4 PORTING MM5 IMPLEMENTATION TO WRF

The approach for porting the technique as implemented in MM5 to WRF was performed in several steps. This task involved the modifications in WRF modeling system (preprocessors as well as the model itself) and modifications to the tools processing GOES observations and model predictions.

The first step as described in the task description was to suppress convective parameterization in the areas where the model was over-predicting clouds. In the previous MM5 studies convective suppression had greatly improved model statistics. Therefore, convective suppression was considered as the low hanging fruit in this process. However, the implementation of convective suppression did not yield the desirable outcome as the response from WRF was different from that of MM5. WRF compensated for the lack of subgrid clouds by creating grid resolved clouds. Upon closer examination of the results it was realized that the way WRF responded to convective suppression was perhaps more plausible. Details of this work are documented in the following section as well as in Appendix A. Appendix A describes the technique and modifications to the code in detail. It also documents the results from a winter test case study for February 2008. To further test the technique for a summer case study, WRF simulations with convective suppression were repeated for August 2006. These results are presented in the following sections.

A major component of this work is to establish a target vertical velocity (lifting or subsidence) to create and sustain clouds in the areas of under-prediction and to remove clouds in the areas of over-prediction. As implemented in MM5, target vertical velocity is estimated based on statistical relationships between model internal variables. These relationships are arrived at by multivariate regression using IMSL statistical libraries (http://www.roguewave.com/products/imsl-numerical-libraries.aspx). A Fortran code developed for this purpose was used to create the target vertical velocities needed to bring model in agreement with observation. The code uses observed cloud fields (albedo and cloud top temperature/pressure) and based on model statistics estimates a target vertical velocity, the altitude of the target vertical velocity, and the layer thickness where this vertical velocity must be sustained.

Target vertical velocities and their altitudes are then passed to divergence adjustment software that uses variational technique to create three-dimensional wind components needed to achieve and sustain target vertical velocities. The detail description of this software and its use is documented in Appendix B. The new horizontal components of wind field are then used as the nudging fields for WRF.

The following section documents the results from convective suppression task for the summer of 2006 followed by a brief description of the variational code. The details of its implementation within WRF and the results from another case study for the winter of 2008 are described in Appendix A.

4.1 Overall Summary of Convective Suppression Run

This activity was part of task 2 in the proposal work plan: "Evaluation of model performance including only a suppression of the convective parameterizations based on satellite data".

The following summarizes the results of implementing the suppression of convection in areas where GOES data indicated clear conditions for a WRF 36-km run from 00 UTC 5 June 2006 through 12 UTC 28 September 2006 using the Kain-Fritsch cumulus parameterization. The run not using the GOES data will be referred to as the "control run", whereas the run using the GOES data will be referred to as the "assimilation run". Spatial plots comparing the control and assimilation runs use the colors of Table 2.1 showing the spatial extent of over- and under-predictions as well as clear/cloudy areas that are in agreement. Also, it should be noted that the definition of clouds for comparison in this section is based on cloud top pressure only and does not rely on cloud albedo.

As documented in Appendix A, in this part of the activity model columns having precipitable cloud water (PCW) values greater than 6.77 x 10^{-3} mm were considered to be "clouds" for the comparison with the satellite data. From the satellite perspective, clouds on the 36-km grid are defined by having cloud cover percentages greater than or equal to 33%. Based on these definitions of model and satellite clouds, a contingency table can be defined as shown in Table 2.1, where "H" is the number of grid cells where both model and satellite indicated cloud (blue, category one or "CAT-1"), "F" is the number of grid cells where the model indicated clouds and the satellite indicated clear conditions (green, category two or "CAT-2"), "M" is the number of grid cells where the model had clear conditions and the satellite indicated clouds (orange, category three or "CAT-3"), and "Z" is the number of grid cells where both model and satellite indicated clear conditions (red, category four or "CAT-4"), Where it is appropriate this same color scheme will be used

Table 4.1. Contingency table categories for model-satellite cloud comparison.

	Satellite Cloud	Satellite No-Cloud
Model Cloud	H	F
Model No-Cloud	M	Z

in all plots. Based on these four cell counts, an agreement index is defined by (4.1). This index will be one of the measures used in evaluating the assimilation run. Model cloud top heights utilize a threshold value of the cloud water mixing ratio, which for these calculations was set at 10^{-3} g kg^{-1}. Model cloud top heights are determined by first determining the first two adjacent model layers, from the top down, where the top layer has a cloud water mixing ratio below the threshold, and the bottom layer has a cloud water mixing ratio at or above the threshold. The cloud top height is then calculated assuming a linear relationship with height between these two layers and set equal to the height where the cloud water mixing ratio is equal to the threshold value. Satellite cloud top heights were determined by finding the two adjacent assimilation model levels which bracketed the GOES cloud top pressure field and then hydrostatically determining the level which had a pressure equal to the GOES cloud top pressure value.

In Appendix A the issues surrounding the shallow convection mode within the Kain-Fritsch scheme were discussed. In that preliminary test run, when all convection was terminated when the GOES observations indicated clear conditions, then an erroneous high bias in low clouds in maritime areas was introduced. In these runs the partial compromise introduced in Appendix A was used. If the GOES observations indicated clear conditions, then deep convection was not allowed, but the shallow convection component of the Kain-Fritsch scheme was allowed.

$$AI = \frac{H + Z}{H + M + F + Z} \tag{4.1}$$

Figure 4.1 shows the cloud top percentages versus cloud top height classes for the control WRF run, the GOES assimilation WRF run, and the GOES observations for the period from 00 UTC 5 June 2006 through 12 UTC 28 September 2006, all on the 36-km grid. The first observation is that the control and assimilation runs are very similar, with the bin percentage values within about 1-2% of each other. The second observation is that the control and assimilation runs have considerably more high clouds in the 12-km, 14-km, and 16-km bins than the GOES observations, and conversely, have much fewer clouds compared to the GOES observations in the 2-km, 4-km, 6-km ,and 8-km bins.

Figure 4.2 shows the observed GOES cloud top height percentages versus observed cloud top height classes for those grid points and times when the assimilation model cloud top heights were in the 12-km, 14-km, and 16-km bins. It shows that a broad range of observed cloud heights corresponded with those situations, but with over half being in the 2-km, 4-km, 6-km, and 8-km observed bins. Figures 2.1 and 2.2 seem to indicate that the cumulus parameterization is creating too many deep clouds, and with respect to its shallow convection component, not enough shallow clouds. When cumulus parameterizations create clouds which are too deep, it can usually can be related, among other things, to the behavior of the updraft parameterization, and/or the degree of instability within the planetary boundary layer (PBL). With regard to the former, updraft dynamics such as updraft entrainment and/or updraft mass flux at cloud base contribute to the height of the convective updraft. This is not meant as a criticism of the Kain-Fritsch scheme - all cumulus parameterizations struggle with these issues. In regards to the latter issue of PBL instability, if the PBL physics scheme and/or the radiation parameterization are creating PBL conditions which are too unstable then the cumulus parameterization will likely produce clouds which are too deep. The lack of shallow convection may be related to the fact that in the current Kain-Fritsch WRF code, the turbulent kinetic energy (TKE) is “hard-wired” as a constant value of 5.0 m^2s^{-2}.

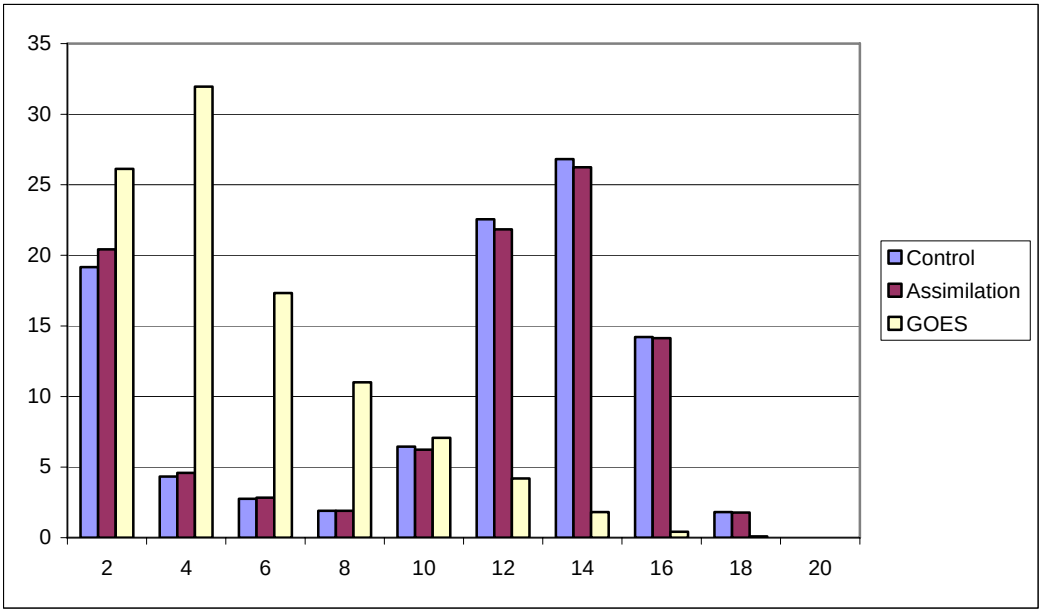


Figure 4.1. Cloud top percentages (y-axis) versus cloud top height classes (x-axis) for the following data sources: 1). control WRF run (blue bars), 2). the GOES assimilation WRF run (violet bars), and 3). the GOES observations (yellow bars). The cloud top height classes (x-axis) are in km with the labels referring to the top end of each 2-km class. For example, the 12-km bin represents cloud top heights greater than 10-km but less than or equal to 12-km. The data cover the period from 00 UTC 5 June 2006 through 12 UTC 28 September 2006. The sum of all percentages for each data source sum to 100%.

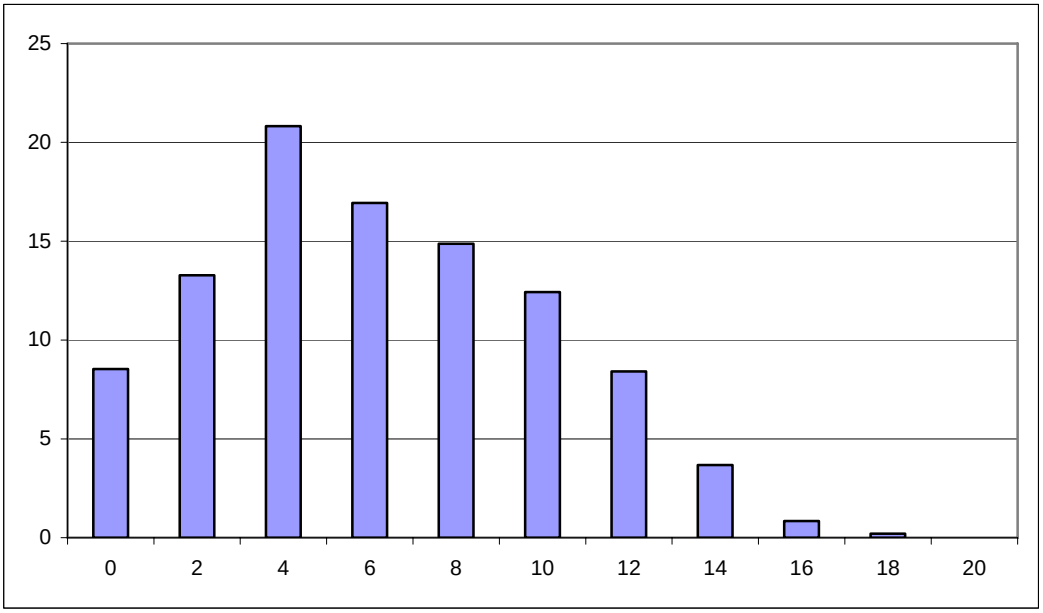


Figure 4.2. Observed GOES cloud top height percentages (y-axis) versus observed cloud top height classes (x-axis) for the GOES assimilation WRF run (violet bars in Fig. 1) for those grid points and times when the model cloud top heights were between 10 and 16 km. As in Fig. 1, the cloud top height classes (x-axis) are in km with the labels referring to the top end of each 2-km class. The "zero" x-axis class contains those situations where the model had clouds but GOES indicated clear conditions. Percentages are relative to the total number of model grid points meeting the 10-16 km cloud top criterion.

Figure 4.3 shows the cloud top height classes for the control and assimilation WRF runs across the cloudy-clear contingency table categories with the exception that the clear category (i.e., the "red" and CAT-4 condition) is not plotted. The first observation is that the largest percentages are the orange bars (cases where the model was clear but GOES had clouds) for observed cloud top heights in the 2-km, 4-km, 6-km, and 8-km bins. This is the similar conclusion drawn from **Figure 4.1**. The second observation is related to the category where the convective suppression could potentially improve results (green bars, cases where the model had clouds but GOES was clear). These percentages were small, typically being in the 1-3% range. Of these situations, about 72% were over water, with almost all of these being over oceanic areas. Because there were so few of these situations the opportunity for improvement was small in regards to the benefit of suppressing convection when the GOES data indicated clear conditions.

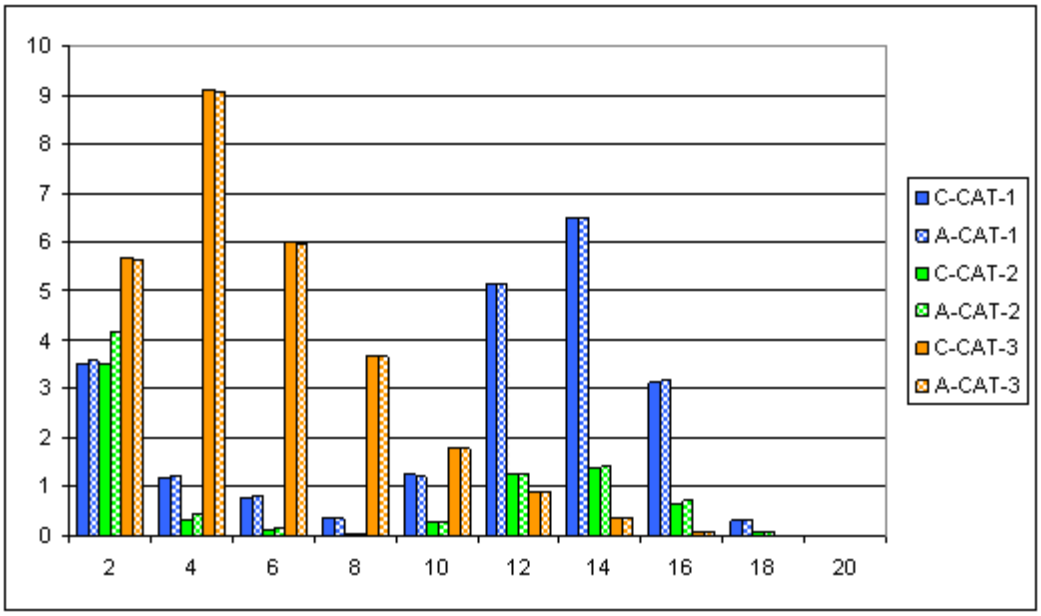


Figure 4.3. Cloud top percentages (y-axis) versus cloud top height classes (x-axis) for the control WRF control and GOES assimilation runs across the cloudy-clear contingency table categories. The cloud top height classes (x-axis) are in km with the labels referring to the top end of each 2-km class. Solid bars refer to the control run, whereas shaded bars refer to the assimilation run. The colors follow the same logic as in Table 1. Blue refers to cases where the model and GOES had clouds. Green refers to cases where the model had clouds but GOES was clear. Orange refers to cases where the model was clear but GOES had clouds. The case where both the model and GOES had clear conditions is not plotted. The percentage of clear points for the control run was about 43%, and about 42% for the assimilation run.

4.1.1 Time Series of Agreement Index

Figure 4.4 and Figure 4.5 are plots of the hourly agreement index for the control run (dark blue curve) and the assimilation run (violet or pink curve) for the period 0000 UTC 05 June 2006 through 1200 UTC 28 September 2006. The control and assimilation runs are very similar with the control run being slightly better (higher values of the agreement index) than the simulation run. The sinusoidal pattern each day arises with relatively high values in the morning before convective cloud development, and lower values in the afternoon when convective cloud development is maximized. Two long periods had no GOES cloud top pressure data. One was from 12-26 July (Julian days 193 through 207), and the other was from 9-15 August (Julian days 221 through 227). The agreement index has values which typically run between 0.50 and 0.75.

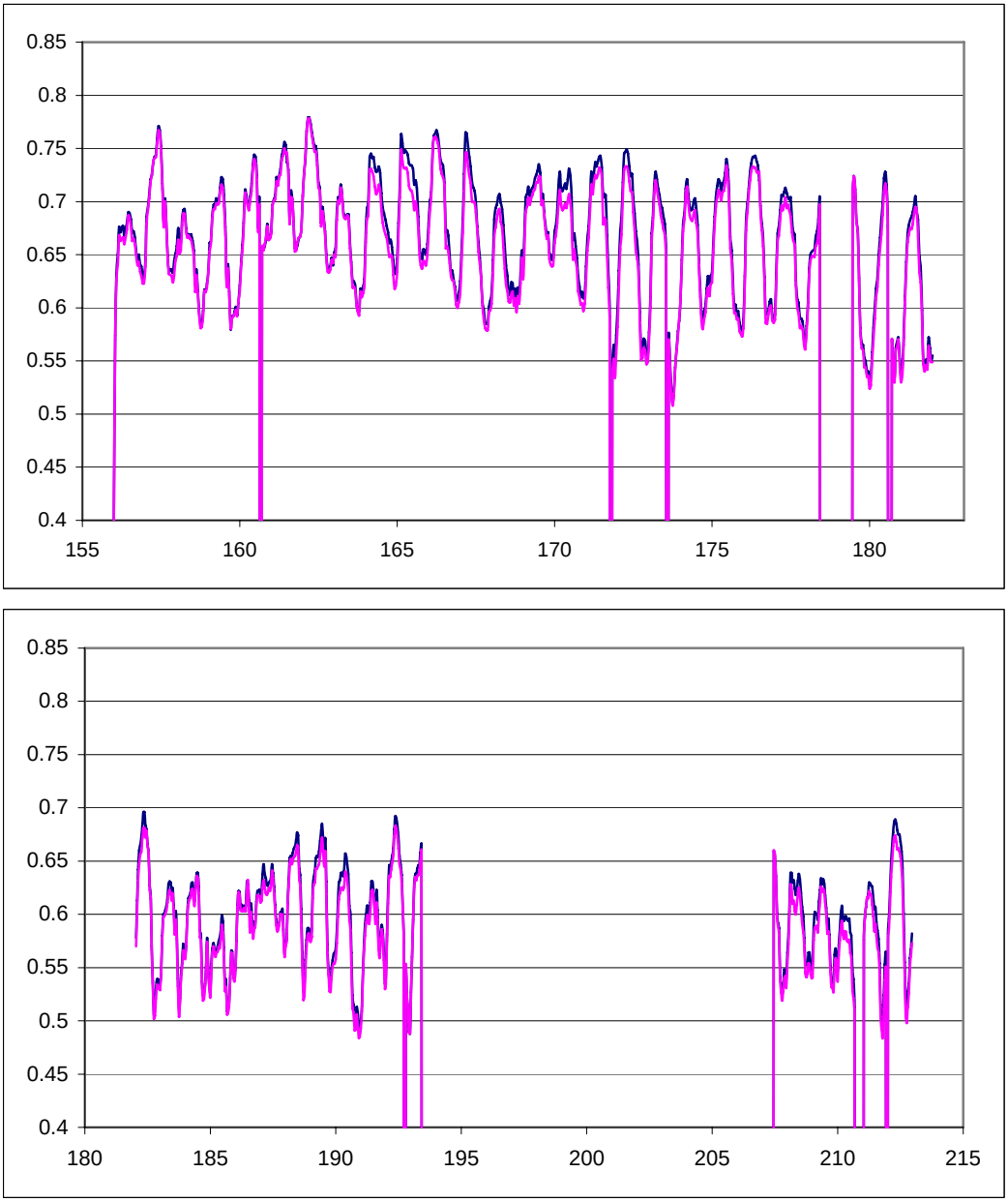


Figure 4.4. Agreement index (y-axis) versus Julian days (x-axis) for the control WRF run (blue curve) and the GOES assimilation WRF run (violet curve). Top panel is for June 2006. Bottom panel is for July 2006. Values below 0.40 are periods where GOES data were missing. Data frequency is hourly.

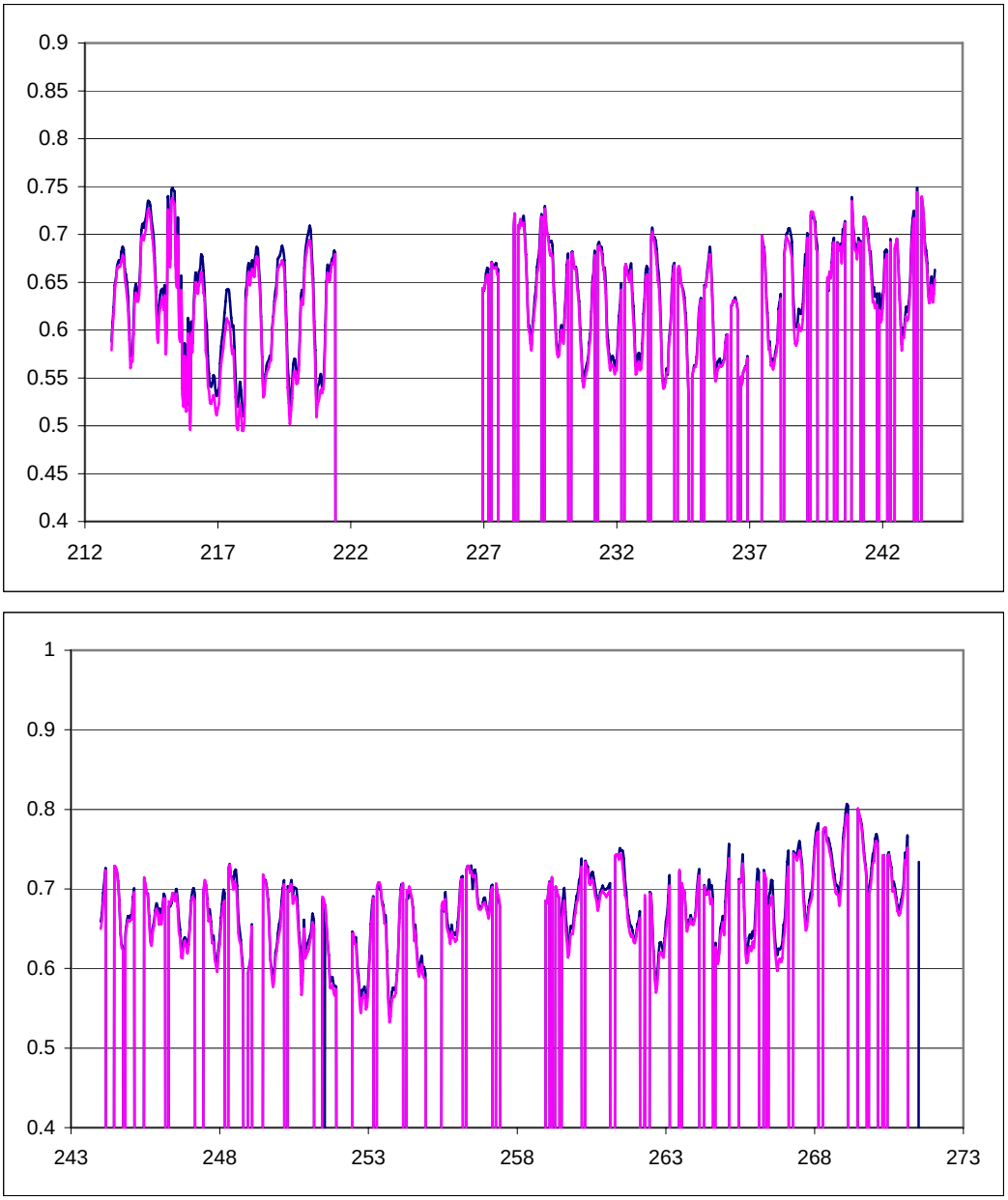


Figure 4.5. Agreement index (y-axis) versus Julian days (x-axis) for the control WRF run (blue curve) and the GOES assimilation WRF run (violet curve). Top panel is for August 2006. Bottom panel is for September 2006. Values below 0.40 are periods where GOES data were missing. Data frequency is hourly.

4.1.2 Plots for 1800 UTC 9 July 2006

Figure 4.6 Figure 4.7 give three plots as a specific hourly example of some the statistics previous described for the entire model runs. The date of 1800 UTC 9 July 2006 was chosen because it had the lowest agreement index for the simulation run for the hours between 1800 and 2300 UTC and for the period 0000 UTC 06 June 2006 through 0000 UTC 28 September 2006. The top panel of Figure 4.6 shows the contingency table colors (from Table 4.1). The most prevalent error color are the "orange cells" which correspond to the situation where the model had clear

conditions and the satellite indicated clouds. The bottom panel of Fig. 6 shows the cloud top heights for these same cells. A broad range of heights are included in the "orange category". Figure 7 shows the GOES insolation for the same time. It shows that the insolation values are significantly reduced for most of the locations in the "orange category".

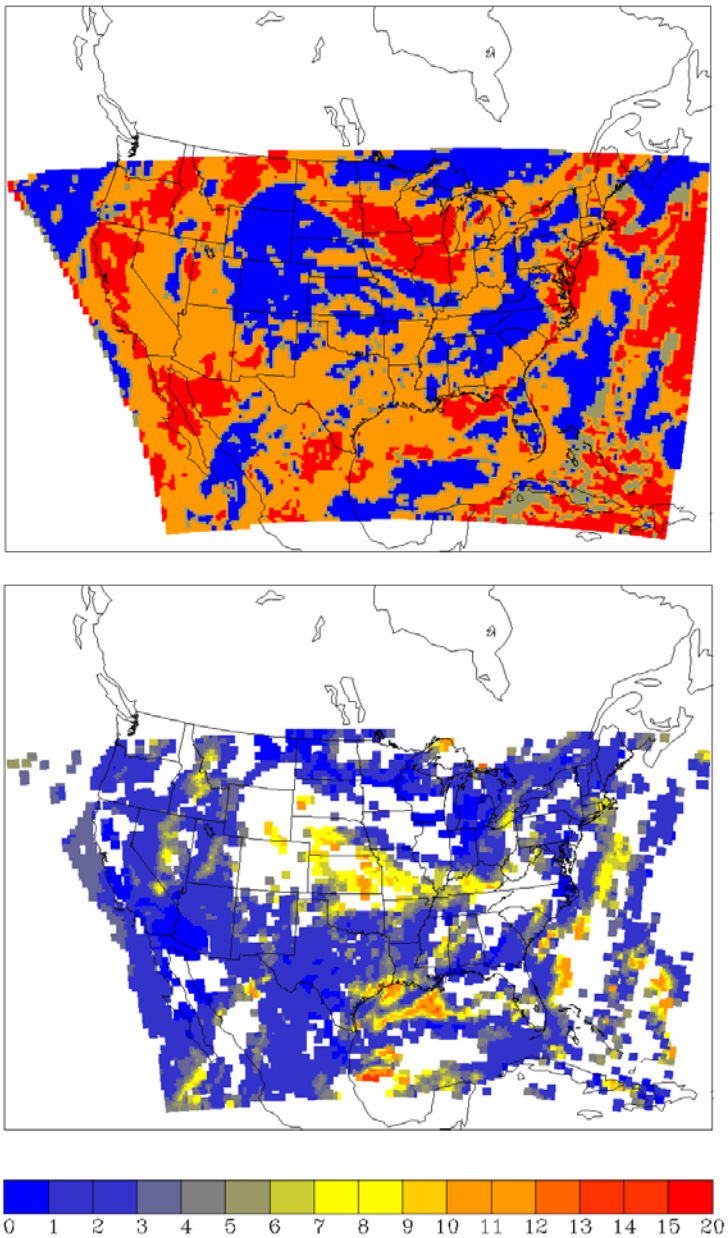


Figure 4.6. Plots are for the time 1800 UTC 9 July 2006 for the simulation run. Top Panel: Colors correspond to the contingency table in Table 2.1. Blue cells are where both model and satellite indicated clouds, green cells are where the model indicated clouds and the satellite indicated clear conditions, orange cells are where the model had clear conditions and the satellite indicated clouds, and red cells are where both model and satellite indicated clear conditions. Bottom Panel: Simulation cloud top heights in km for the "orange cells" in the top panel.

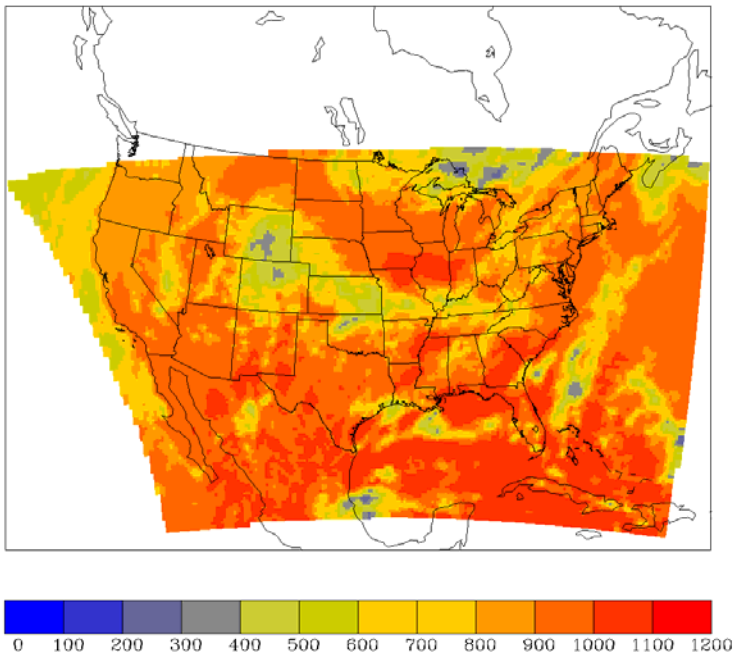


Figure 4.7. GOES insolation in W m^{-2} for the time 1800 UTC 9 July 2006.

4.1.3 Conclusions

The convective suppression simulation run had error statistics almost identical with the control run. The opportunities for this approach to be beneficial were very limited in this run. This was partially due to the dry pattern for much of the United States up until September.

Given the documentation of the issues related to the shallow-convection mode of the Kain-Fritsch cumulus parameterization scheme, it raises the question whether it would be worth the time and effort to switch to a PBL scheme which has a prognostic field of turbulent kinetic energy and making the necessary subroutine argument changes in the cumulus parameterization scheme to make that information accessible to the shallow convection calculations. This will also make the implementation more complicated and therefore will not be considered as an imminent task.

Figure 4.8 provides a schematic of the interrelationships of the two approaches currently being employed to assimilate clouds into the WPS/WRF system. Approach one builds on the basic methodology used in the old MM5 simulations which uses a set of multiple linear regression equations to establish vertical velocity targets and other needed parameters. Approach two, utilizes nudging and/or the 3DVAR package available in WPS/WRF system. This approach also has been tested and the results will be presented in the following sections. Approach two utilizes many of the tools developed for the first approach and therefore avoids repetitions.

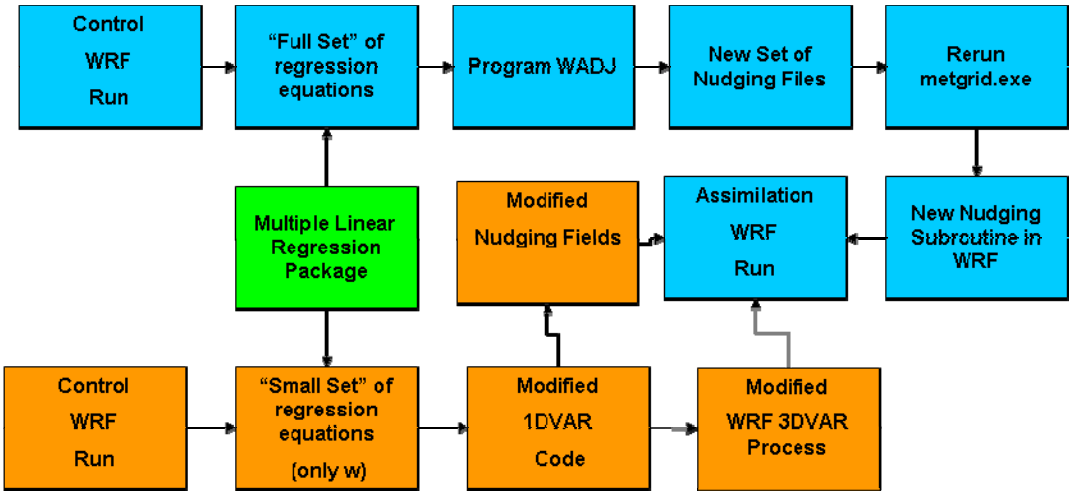


Figure 4.8. Current status of two approaches of cloud assimilation work. Color and brief description: (blue; path 1, transitioning/streamlining MM5 approach to WPS/WRF system), (orange; path 2, simplified approach using the tools from path 1), (green; statistical package used in both approaches).

4.2 One-Dimensional Variation Technique

A key component of the current technique is the one-dimensional variation approach for adjusting horizontal components of winds in order to support the target vertical velocity. The technique follows the approach suggested by O'Brien (1970). Given the target vertical velocity and elevation, the approach minimally adjusts the horizontal divergence fields in a column to achieve the target. Then, the three-dimensional wind field is adjusted to preserve the mass and continuity. Appendix B documents the details of this approach by describing the theoretical basis of the technique as well as technical details of its implementation and use.

5 ALTERNATE APPROACH

5.1 Purpose and Strategy

The overall goal of this project was to have a cloud assimilation technique that will be used in an operational setting. Thus, operational concerns such as ease of use and computational efficiency were also important factors for consideration. A major objective of the project was to recommend a way forward for achieving this objective. Thus, in consultation with TCEQ, while continuing to streamline and implement the technique as it was incorporated in MM5, we also revisited our basic assumptions and took a new look at the whole problem. This led to identifying the key variables that needed to be adjusted in the model and also in simplifying the implementation and thus the operational use of the technique.

The basic approach has been to use a GOES cloud image to determine the cloud truth. Therefore, the strategy was first to examine differences between model clouds and the GOES clouds. Then, adjust key model variables such as vertical velocity and relative humidity so that the model would produce clouds where the GOES image has clouds and to suppress model clouds where the GOES image says it is clear. Therefore, a natural starting point was to develop statistical relationships in the model for adjusting vertical velocity and relative humidity.

The following documents the activities in this regard that resulted in an alternate approach for cloud assimilation. This approach uses many of the same tools that were used in the previous approach, but the simplicity of the approach makes it more attractive for operational use. The preliminary results from this approach yields improvements in model performance that are comparable to the previous approach. We have already identified many areas where the alternate approach can be improved and will be followed in the future studies.

5.2 Adjustment of the Model Environment to Support Clouds

The initiation and assimilation of clouds in weather forecast models has been the subject of many investigations. Yet, evidence suggests that there are still major errors in cloud placement. This is especially true at the spatial scale at which air quality models operate. It is particularly frustrating in air quality SIP modeling since they are after the fact runs that the observed cloud field is known from satellite observations but models have significant differences in cloud placement. At UAH considerable attention has been given to replacing model cloud transmissivity with satellite observed transmissivity (McNider et al 1995, Pour-Biazar et al 2007) in air quality models. These previous activities that directly replaced model transmissivity and cloud tops with satellite observations provided improvements in model performance. However, it produced a physical inconsistency in the model system. Insolation and photolysis fields derived from satellite data did not agree with the model clouds. Thus, locations in the model where deep convection or cloud venting of the boundary layer was occurring were not consistent with the locations where the satellites indicated clouds were located. Additionally, cloud water that would impact long wave radiation or chemistry was in the wrong location.

Rather than adjusting model transmissivity it would be preferred to insert cloud water into the model at locations where the satellite indicates clouds. Previous attempts at using satellite data to insert cloud water have met with limited success. For example, Lipton and Modica (1999) used GOES-7 data to adjust the model relative humidity field in stratiform cloud areas and found a general improvement in the model simulation but only for about 6 hours.

The problem is that cloud water typically depends on a water vapor and temperature environment to provide the relative humidity to sustain the cloud liquid water. Conversely, when liquid water is removed from the model where observations show no clouds, the model will continue to produce new water. Direct insertion of liquid water can even deteriorate model performance. As an example, attempting to insert clouds that satellites show at a position where the model is clear means that you are likely inserting clouds where the model has subsidence (broad-scale downward motion) as opposed to lifting. Inserting water in this situation where the model has subsidence will cause evaporation and further subsidence, exactly the opposite of supporting the clouds that the satellite observes. Yucel et al (2003) discovered that adjustment of the model dynamics and thermodynamics was necessary to fully support the insertion of cloud liquid water in models.

In reality, the issue with supporting clouds in models goes beyond thermodynamic support. For clouds to persist they must have dynamical support through upward vertical motion. This has been recognized in the weather forecasting community and investigators with NOAA seeking to produce improved initialization have inserted vertical motion in models where clouds were observed but not supported (Albers et al. 1996). However, these motions were relatively ad hoc and the inserted vertical velocities relatively small. The results, though they produced some improvement, had limited success in changing the cloud statistics after several hours.

In principal the problem of providing the coincident thermodynamic support and dynamical support could be provided by four-dimensional variational assimilation (4DVAR). 4DVAR employs a strategy that develops local linear approximations of relationships among all variables (the tangent linear approximation) (Lopez, 2006). These partial derivatives among all the state variables (the Jacobian matrix) define the needed relationships so that required changes in humidity and vertical motions could be found if liquid water were changed from satellite observations. Variational minimization is coincidently used to reduce the difference between observations and model values. The development of the Jacobian can be done using tools from non-linear analysis (McNider et al. 1995b) or by running the model in a forward/backward mode to determine relationships among the variables. However, the application of 4DVAR to cloud initialization has been limited by the inability to define the Jacobian for cloud processes. A survey of the state of science reveals a lack of success in developing 4DVAR for clouds except for simple representations in very coarse grid models. This is in large part because cloud processes and cloud initiation are highly non-linear. This is further exacerbated by the fact that clouds in models are highly parameterized and the relationships have many conditional and on-off switches which make developing the required inter-parameter relationships difficult if not impossible (Mu and Wang, 2003).

In the present activity we will employ a technique originally started under the NASA GEWEX activity but with further support by NSF and MMS. This technique is basically a statistical

approach to 4DVAR that attempts to develop relationships between satellite-derived cloud properties and targeted variables internal to the models such as grid scale vertical velocity. It also directly confronts some of the parameterization issues by turning off convective parameterizations when satellite observations indicate clear conditions. As mentioned above, the air quality problem is generally different from the forecast problem in that observed clouds are available during the entire period of interest not just at the initialization time. In effect the technique proposed here is an engineering attempt to provide the dynamical and thermodynamical support needed to sustain or clear observed clouds during the air quality simulation.

5.3 Development of Cloud and Satellite Relationships

Unfortunately, at the present, satellites do not provide a good temporal and geographical estimation of cloud water which is needed in the model. However, the Cloudsat suite of instruments (Stephens et al. 2002), on a polar orbiting platform, are beginning to provide limited snapshots of cloud water.

Geostationary satellites provide an excellent geographical perspective on the reflected short wave radiance of clouds. This reflected radiance is close to the on ground perception of the attenuation of short wave by clouds at the ground (Diak and Gautier 1983). Thus, previous work by UAH (McNider et al. 1995a, McNider et al. 1994 and Pour-Biazar et al 2007, have utilized the Diak and Gautier 1983 technique to infer a near inverse of reflectance to specify a cloud transmissivity to correct insolation values and photolysis fields in models.

There has been work in utilizing GOES reflected radiance to infer a liquid water path (Modica and Lipton 1999) for insertion in models. However, a small amount of liquid water can saturate the visible channel so that the dynamic range of retrieved liquid water is limited.

Models generally employ a relationship between liquid water and visible reflectance (many use Stephens 1978) as they attempt to specify radiative properties in the model such as insolation. As a by-product of this process the cloud albedo can be retrieved in the model. **Figure 5.1** shows the internal model relationship between cloud albedo and cloud liquid water. This logarithmic behavior is consistent with the Stephen 1978. Small amounts of liquid water cause a jump in cloud albedo and albedo becomes less sensitive as large volumes of water are present. **Figure 5.2** shows a map of the logarithm of cloud water over the whole domain. The log of the liquid water is roughly related to the linear cloud albedo.

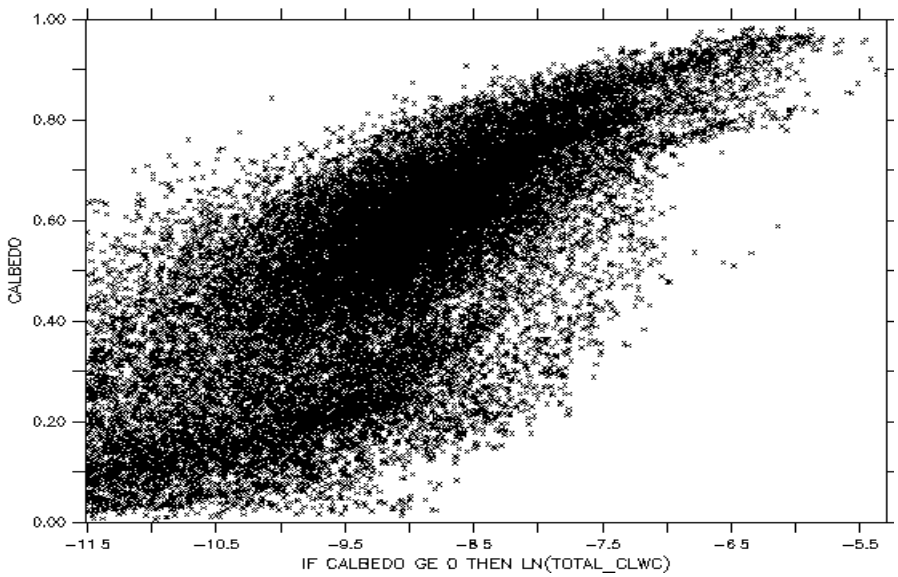


Figure 5.1 Scatter plot of cloud albedo versus total cloud liquid water

In the present investigation we will use the model estimates of insolation to compare against GOES retrieved insolation by the Diak and Gautier (1983) method. We then normalize the insolation by the maximum clear sky insolation from both the model and satellite insolation to provide an approximate cloud albdeo. This normalized insolation or approximate cloud albedo will be the parameter that we will use to compare model clouds versus satellite clouds.

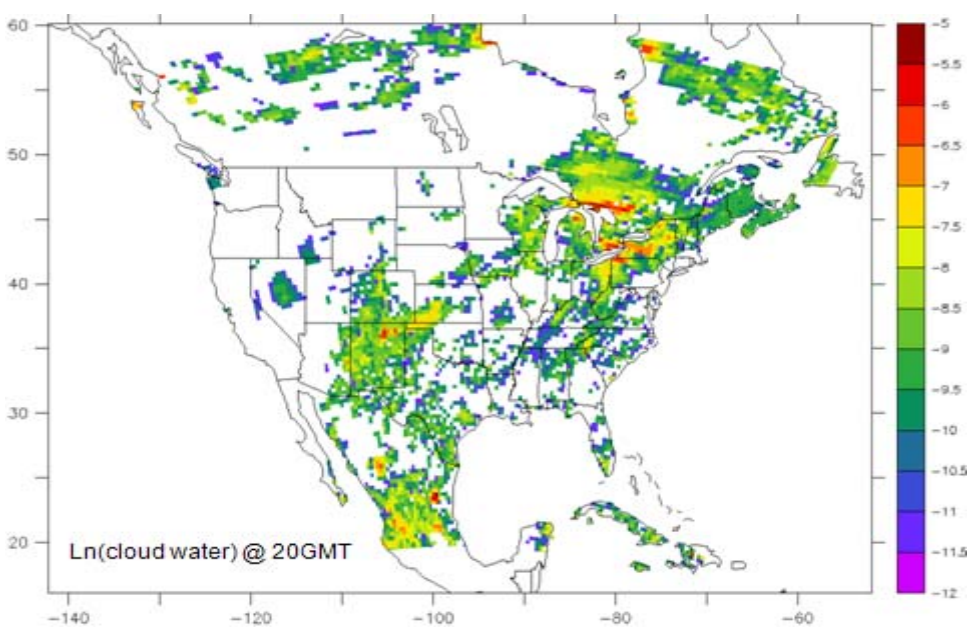


Figure 5.2 Map of the natural log of model cloud water. Note this depiction is close to that of the relative cloud albedo.

5.4 Relationships Between Model Variables and Cloud Water in the Model

In order to adjust the model environment to thermodynamically and dynamically support clouds we need to establish the relationship of clouds to the model variables relative humidity and vertical motion. To do this first raw statistics were calculated and then detailed analyses were made at specific model grid points. **Figure 5.3** shows a map of vertical velocity (maximum w in column) and corresponding cloud albedo at a particular model time.

While there is general agreement that clouds exist where vertical motion is positive there is not a one to one correspondence. This is illustrated in **Figure 5.4** which is a scatter plot of cloud water versus W max. Note that even slight vertical velocities can produce a wide range of liquid water.

Figure 5.5 shows a linear view of cloud albedo versus vertical velocity. Note again that small positive vertical velocities are associated with a wide range of cloud albedo but negative vertical motions are generally associated with no or very thin clouds.

These composite analyses show that we do not appear to have strong functional relations between simple model maximum vertical velocity in a column and cloud albedo. Other parameters appear to be at play. However they do show that even small positive vertical velocities are associated with a range of cloud albedo. To investigate this further a series of points in the model domain were sampled to look at vertical profiles of vertical motion, liquid cloud water and relative humidity. Points were selected in a variety of cloudy and non-cloudy grid points. **Figure 5.6** shows the cloud albedo and points selected.

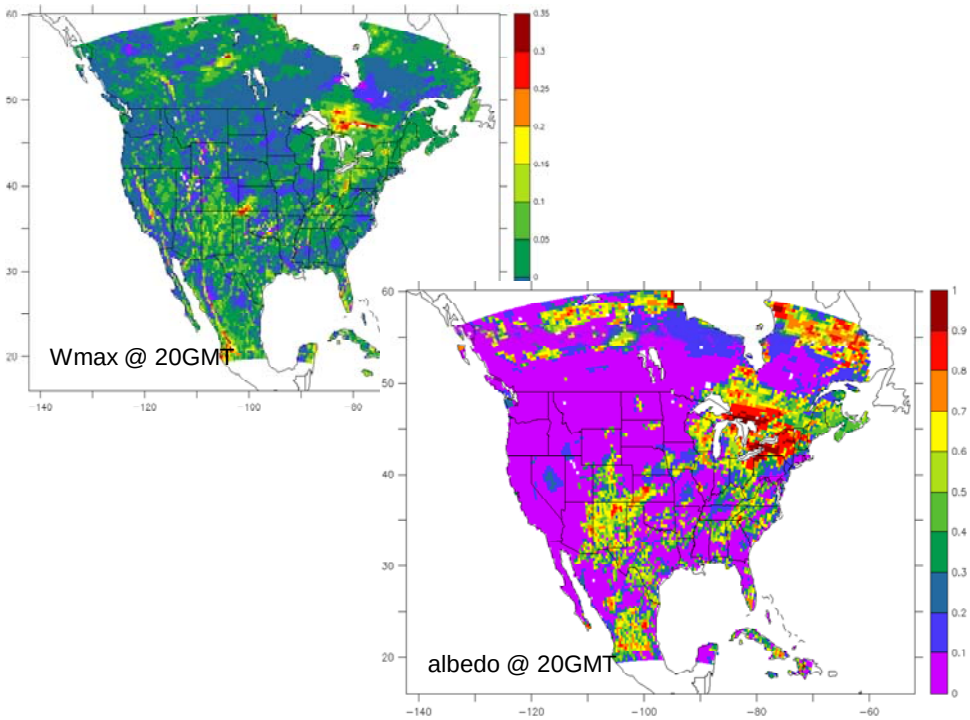


Figure 5.3 Maps of model vertical velocity and cloud albedo for a particular model time.

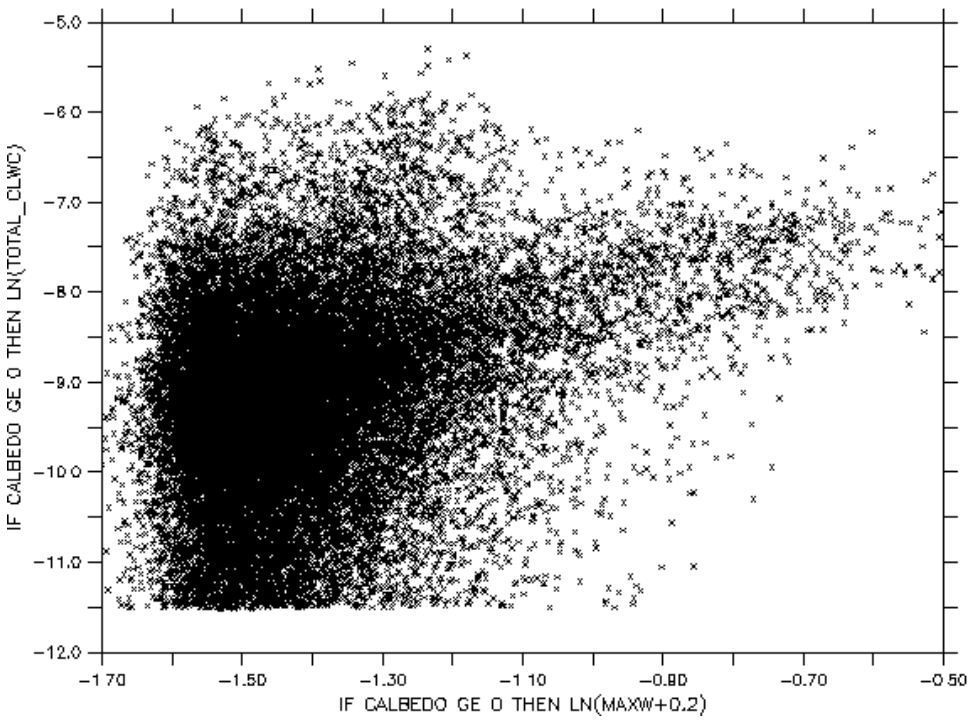


Figure 5.4 Scatter plot of total cloud water and maximum vertical velocity in the model column.

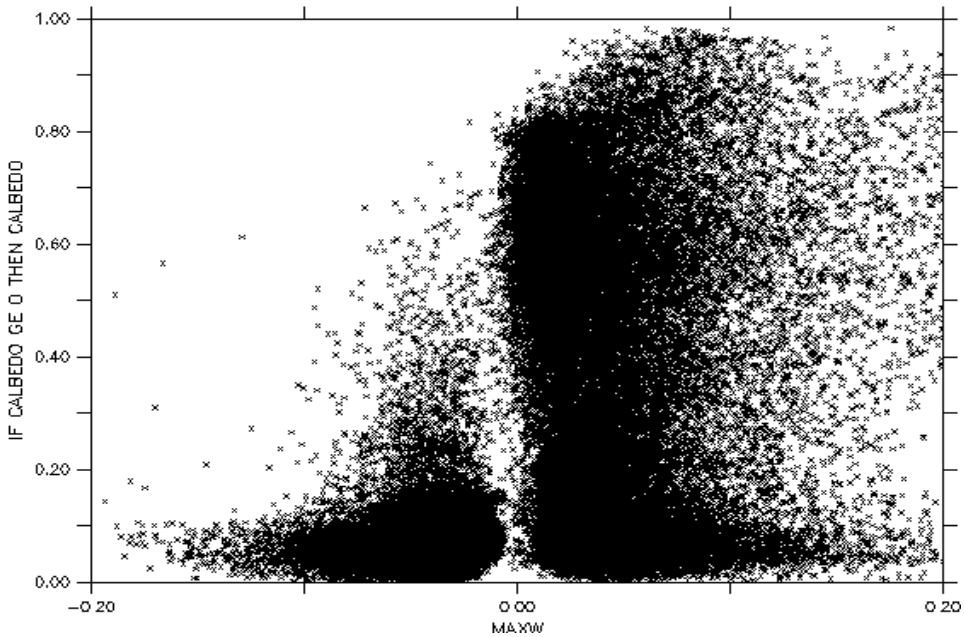


Figure 5.5 Scatter plot of cloud albedo versus maximum vertical velocity in the column.

Nine locations are selected for w, air temperature, and relative humidity profiles. Points are shown as small circles on map. The coordinates are:

- [40,90], [80,90], [115,90]
- [40,70], [60,56], [110,60]
- [40,50], [90,50], [110,45]

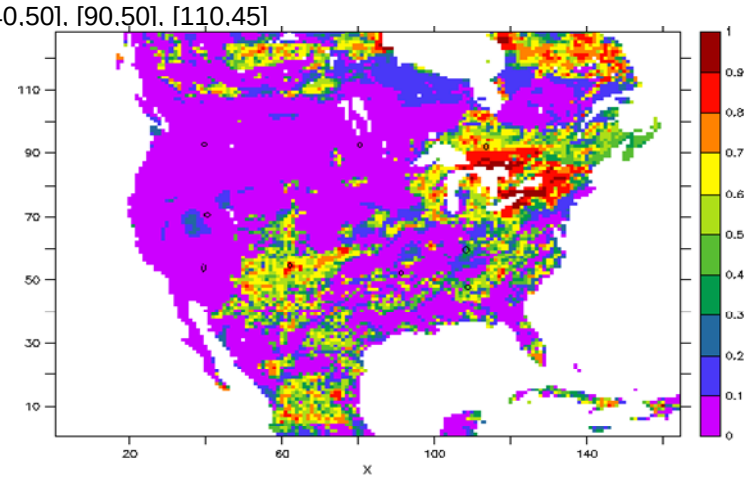


Figure 5.6 Map of cloud albedo and points selected for further analysis. Small circle on map show selected points. Coordinates are given in figure.

Figure 5.7 shows a vertical profiles from the model for a point in a clear area. Note the maximum positive vertical velocity is near the tropopause.(based on the temperature profile). However, the RH (ratio of e/es) is low. So that despite lifting and cooling the air is not brought to saturation. **Figure 5.8** shows profiles for another clear point. Here vertical velocities or negative over the entire column but one area where vertical velocities are near zero show RH above 90%.

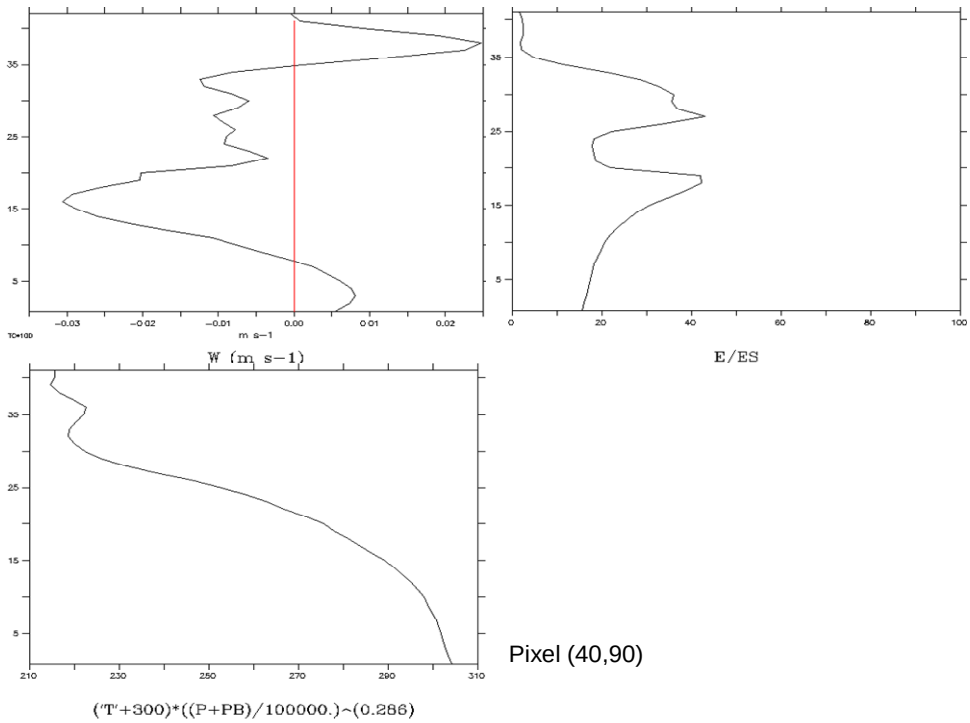


Figure 5.7 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for clear point (40, 90)

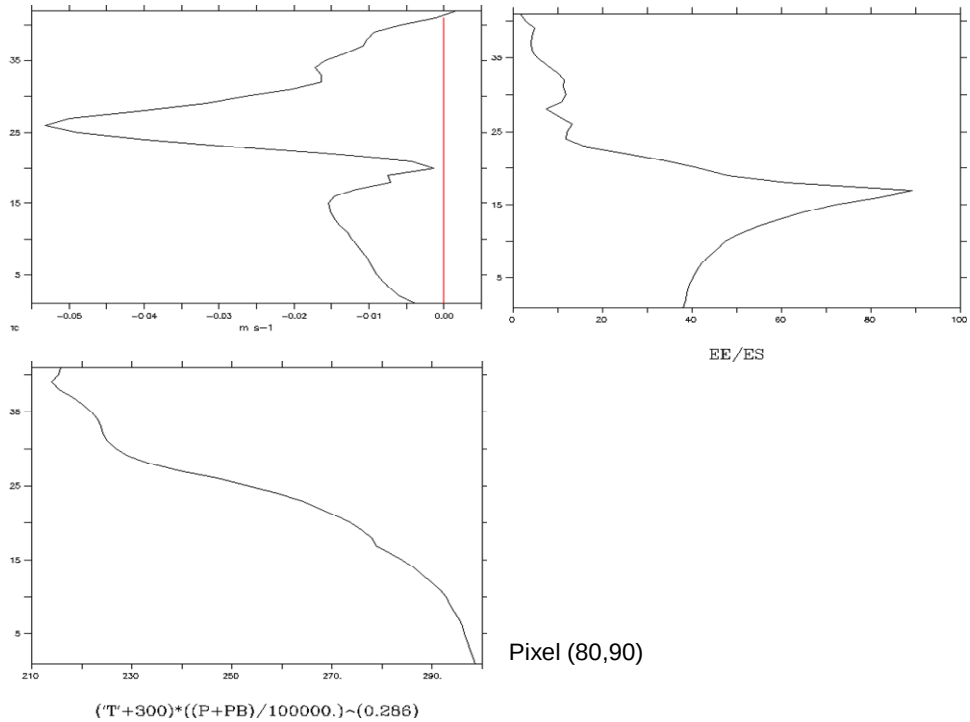


Figure 5.8 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for clear point (80, 90)

Figure 5.9 shows a point that has clouds in the Great Lakes region. Here there is substantial lifting above grid point 20. This evidently produces cooling and RH near 100%. In fact the entire

column is near saturation even in areas with sinking perhaps indicating evaporation of rain in the column. **Figure 5.10** shows another cloudy grid point profile. Here there is substantial upward motion but note that the RH is low. This is a case where the clouds are not being formed because despite lifting the air is too dry. In **Figure 5.11** for point (60,56) there is a broad uplift from the surface up through most of the troposphere. However, the RH only supports clouds near vertical grid point 15 (about 15 meters). Point (110,60) (**Figure 5.12**) is an area of thin clouds. There is substantial vertical motion but only near the tropopause where RH are low. The clouds in this case may be coming from the cumulus parameterization scheme rather than grid scale clouds. **Figure 5.13** shows a grid box that is clear. Here vertical motion is negative and RH is low. **Figure 5.14** shows a cloudy point where upward motion near vertical grid point 20 has evidently produced high RH near the same point.

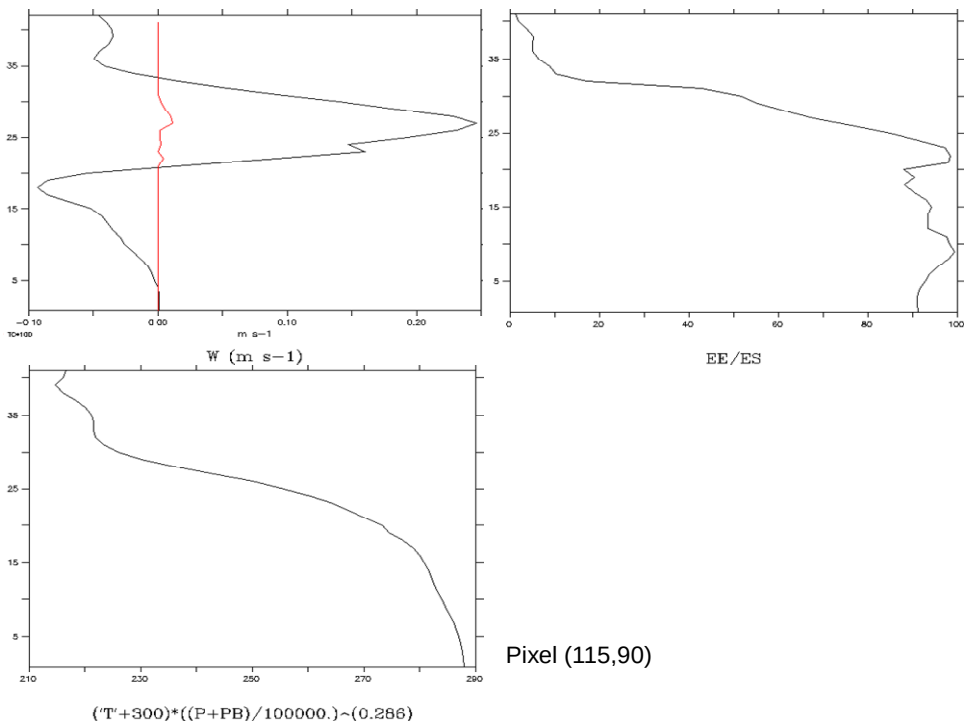


Figure 5.9 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for cloudy point (115, 90)

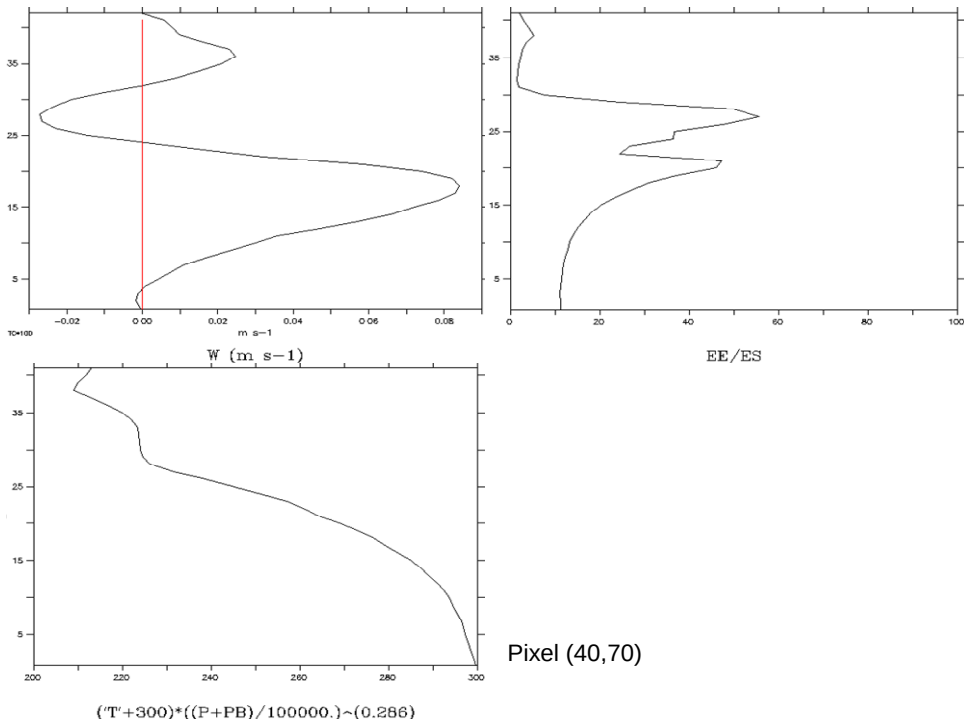


Figure 5.10 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for clear point (40, 70)

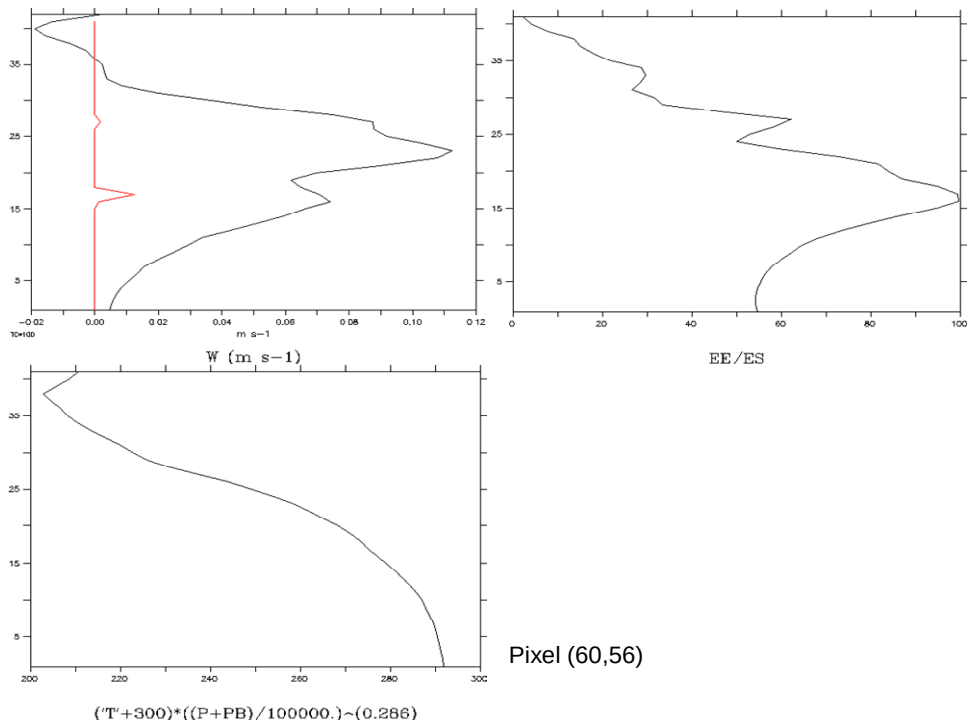


Figure 5.11 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for cloudy point (60,56)

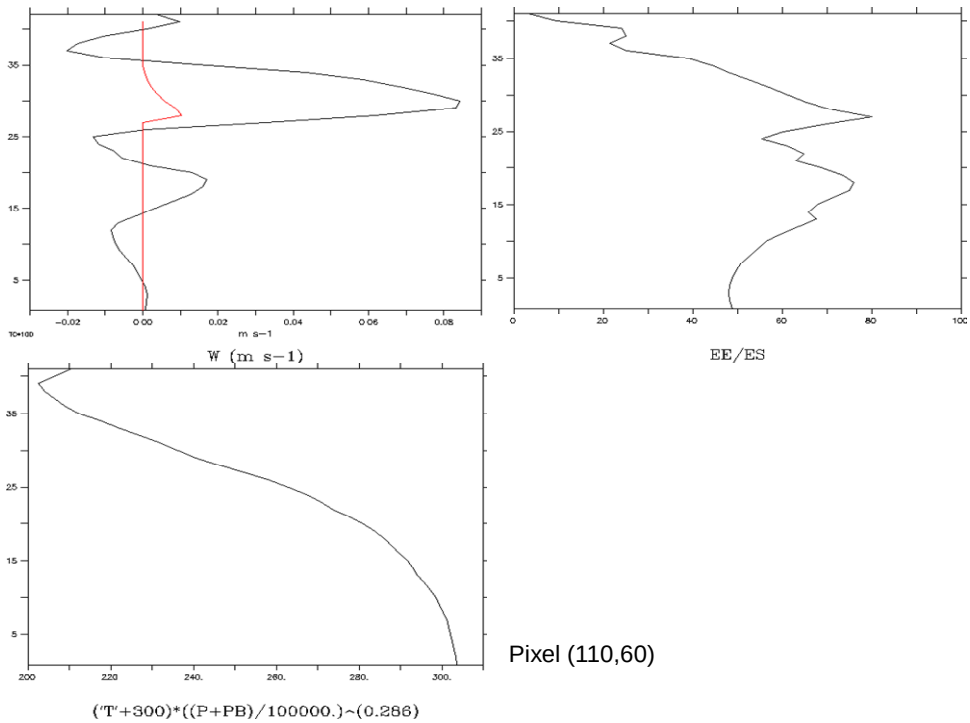


Figure 5.12 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for cloudy point (110,60)

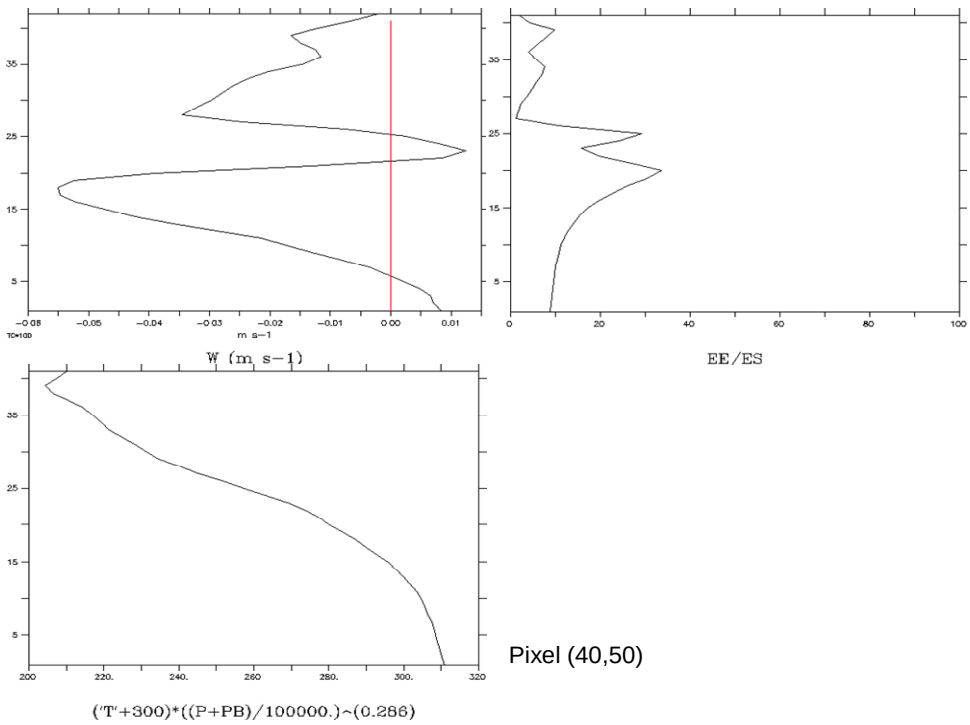


Figure 5.13 Profiles of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for clear point (40,50)

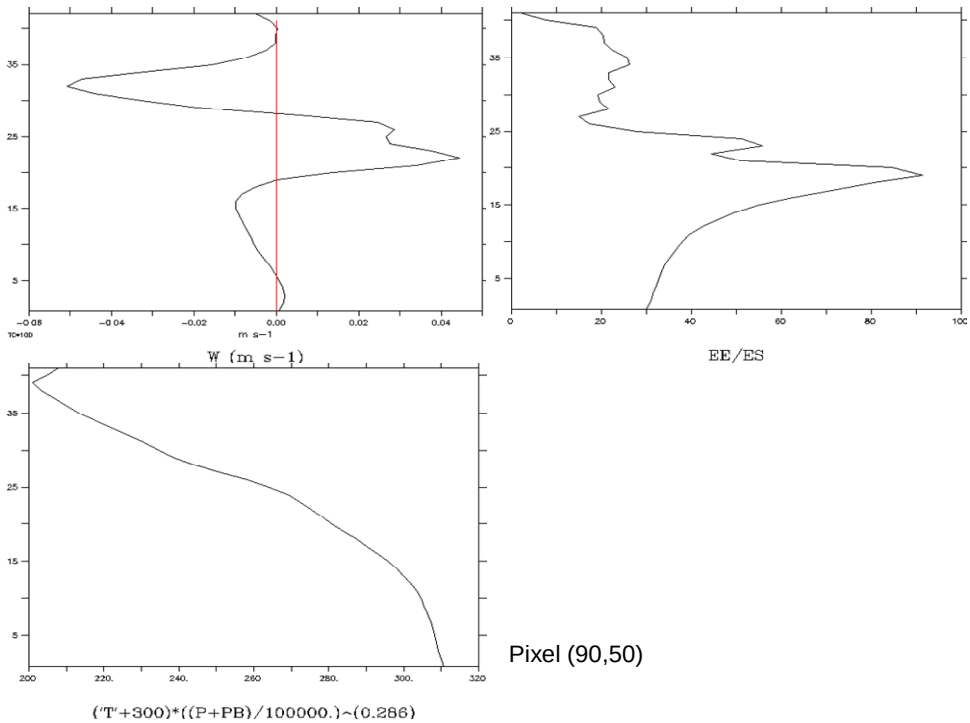


Figure 5.14 of vertical velocity (upper left), relative humidity (upper right) and temperature (lower left) for cloudy point (90,50)

While the raw statistics show very little useful functional relationships with vertical velocity, the analysis of the profiles indicate why simple relationships might fail, it is clear that calculation of the appropriate vertical velocity is tied to vertical position in the column. But, most importantly the vertical velocity must be occurring in area of reasonable moisture for clouds to develop. Further, the raw statistics also show that even small amounts of liquid water can produce clouds and small vertical velocities can produce clouds. Thus it appears that clouds have a very sharp threshold of when clouds form. The following develops an alternative to functional relationships a threshold relationship.

5.5 Alternative Threshold Relations

As mentioned at the start, vertical velocity has a critical role in clouds. Upward motion cools a parcel through adiabatic expansion. This generally brings the air closer to saturation. When starting relative humidities are high clouds will form. On the other hand, subsidence warms a parcel decreasing relative humidity leading to evaporation of clouds. Because the fall velocity of a typical drop is of order 0.10 m/s a droplet will likely fall out of a saturated environment in a matter of hours so that the lifetime of a cloud in the absence of vertical motion is relatively short.

However, the examination of functional relationships above between cloud water and/or cloud albedo with model vertical motion and clouds was not clear. Here we examine the relationship between model clouds and threshold relationships with vertical motion and relative humidity. We take a contingency probability approach, i.e., we examine what is the coincidence of clouds

occurring with vertical positive motion. Alternatively what is the coincidence clouds occurring when vertical motion is negative.

Figure 5.15 shows the contingency relationships for vertical velocity greater than .02 m/s and cloud albedo greater than .1. The image indicates that nearly all the clouds are associated with positive vertical motion, although the functional relationships above showed that the magnitude of cloud albedo was poorly describe by the magnitude of vertical velocity. The bottom right of Figure 5.15 shows that the coincidence of positive vertical velocity and clouds provides a good predictor of cloud albedo.

Figure 5.16 shows the opposite case of where negative vertical velocities are associated with clear areas. It also shows that there is little probability of clouds occurring where vertical velocities are negative.

Figure 5.15 Figure 5.16 together demonstrate the hypothesis that clouds are strongly related to the sign of vertical motion. They also confirm the working view at the start of this path that in a GOES black and white image it is likely that white indicates lifting and dark subsidence.

While vertical velocity is a probable indicator of clouds, relative humidity is also a factor in cloud formation. If a parcel is too far away from saturation even strong vertical velocities may not push the parcel fully to saturation and thus clouds will not form. **Figure 5.17** show RH in the model that can be compared to the clouds and vertical velocity in **Figure 5.15**Figure 5.16.

However, vertical velocity and RH are intimately related e.g. when positive vertical velocities cause lifting and cooling then RH increases. Subsidence yields warming and decreases in RH. RH can also have an effect on vertical velocity. When cloud droplets are in a less than saturated environment then evaporation can cause cooling leading to negative vertical velocities. Also, condensation at supersaturated conditions can lead to buoyancy and lifting.

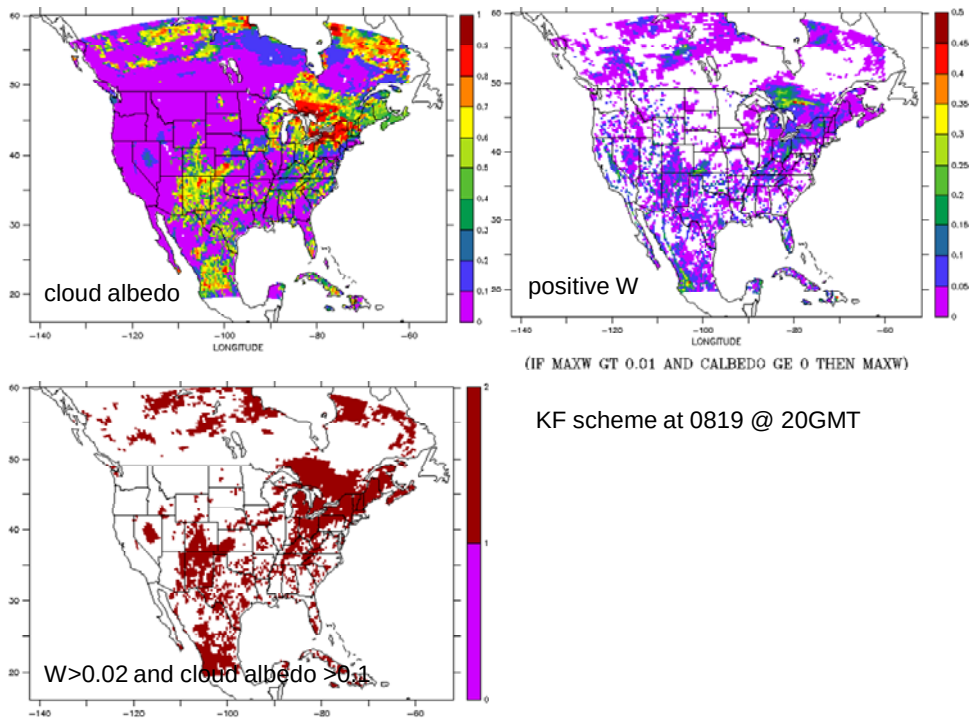


Figure 5.15 Graphical depiction of model relationships. Upper left shows model cloud albedo. Upper right shows model vertical velocity. Lower left shows clouds with contingency of model vertical velocity greater than .02 m/s and cloud albedo >.1.

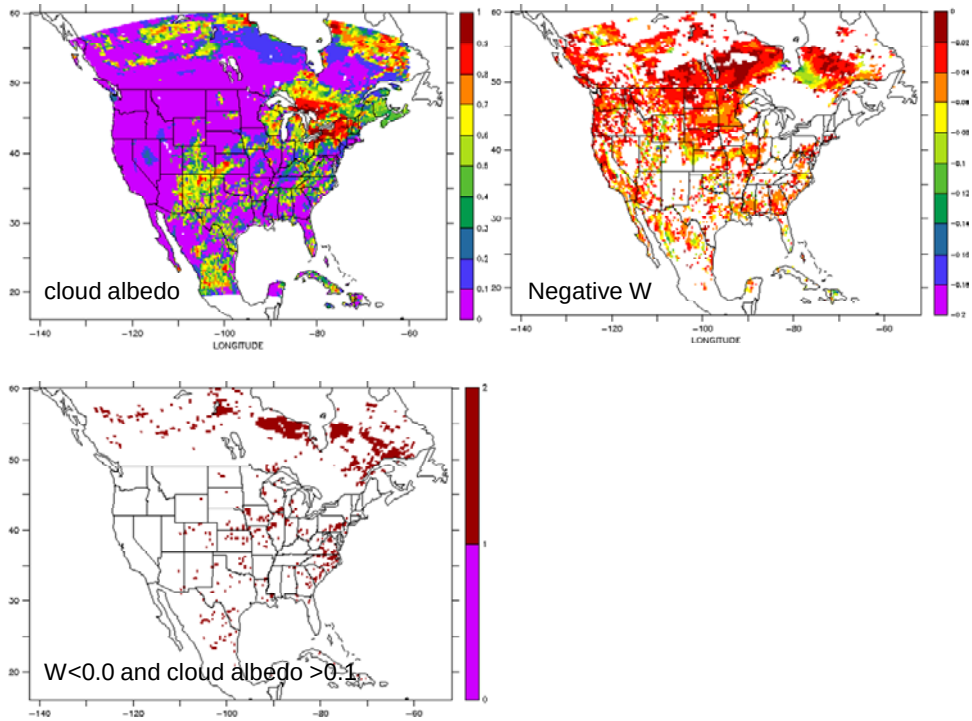


Figure 5.16 Images of model cloud albedo and negative vertical velocity. Bottom left show coincidence of w >0.0 and cloud albedo> .1

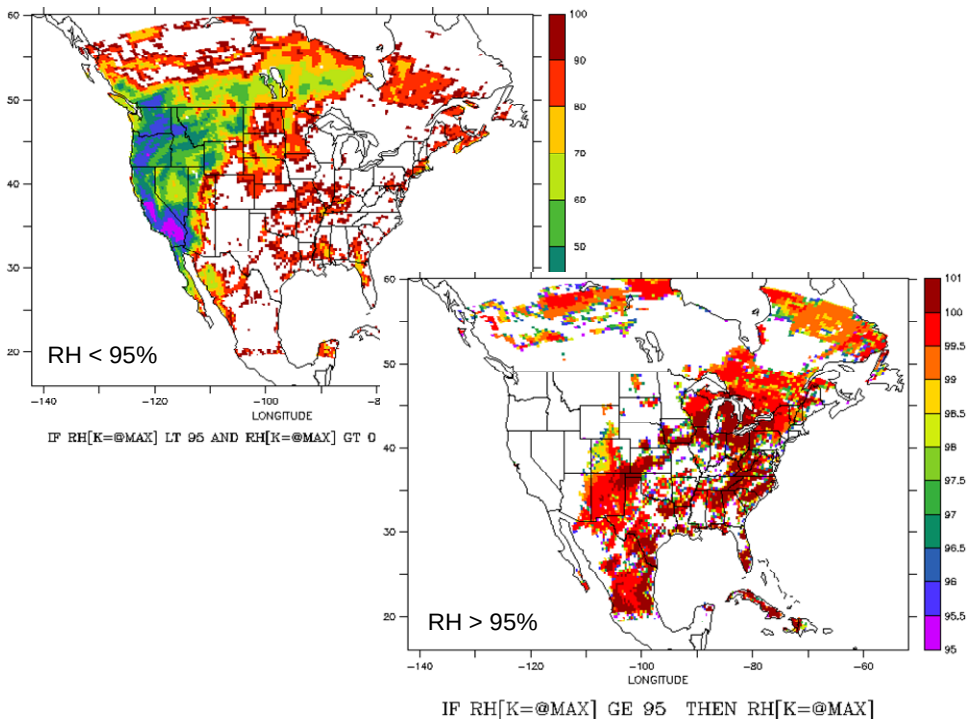


Figure 5.17 Model categorization of relative humidity

5.6 Use of Model Variables to Develop Satellite Contingency Tables

The analysis above indicates that there are coincident conditions (e.g. w>.02, RH>.80) in the model in which clouds are supported and conditions (w< 0 and RH <.50) when clouds are suppressed. Can these conditions be used to determine where clouds might be suppressed are supported by comparing these conditions in the model to where the satellite has clouds? The following examines these contingency relationships in comparison with satellite cloud data first for single events and then for monthly statistics.

Figure 5.18 shows a comparison of the cloud albedo from the model and GOES satellite for August 19 at 2000 GMT. While there are many similarities, there are also many areas of disagreement. Such disagreement can cause considerable changes in solar insolation impacting both temperature and photolysis levels in air quality models. The differences are illustrated in the lower left panel that shows cases where the models has clouds and the satellite doesn't (green pixels) and where the model does not have clouds and the satellite has clouds (orange pixels). Also areas of agreement where the model and satellite both don't have clouds (blue pixels) and where model and satellite have clouds (red pixels). **Figure 5.19** shows a similar depiction for the Grell-Devenyi scheme.

We can now examine the disagreement in light of the model variables (w and RH) which were shown above to have coincident relations with clouds. **Figure 5.20** shows a depiction of pixels (grid points) where the model has clouds and GOES does not. In this case we need to clear clouds from the model. **Figure 5.21** shows where the model has positive vertical motion where

the model has clouds. Thus, the yellow pixels show the places where it may be possible to clear clouds if vertical velocity can be changed from positive to negative.

Figure 5.22 shows the pixels where GOES has clouds and the model does not. This is a case where we need to have the model produce or support clouds. **Figure 5.23** shows the pixels where the disagreement exist and the green pixels show where the model vertical velocities are negative. Thus, they show where we might be able to support clouds if we could impose w >0.0.

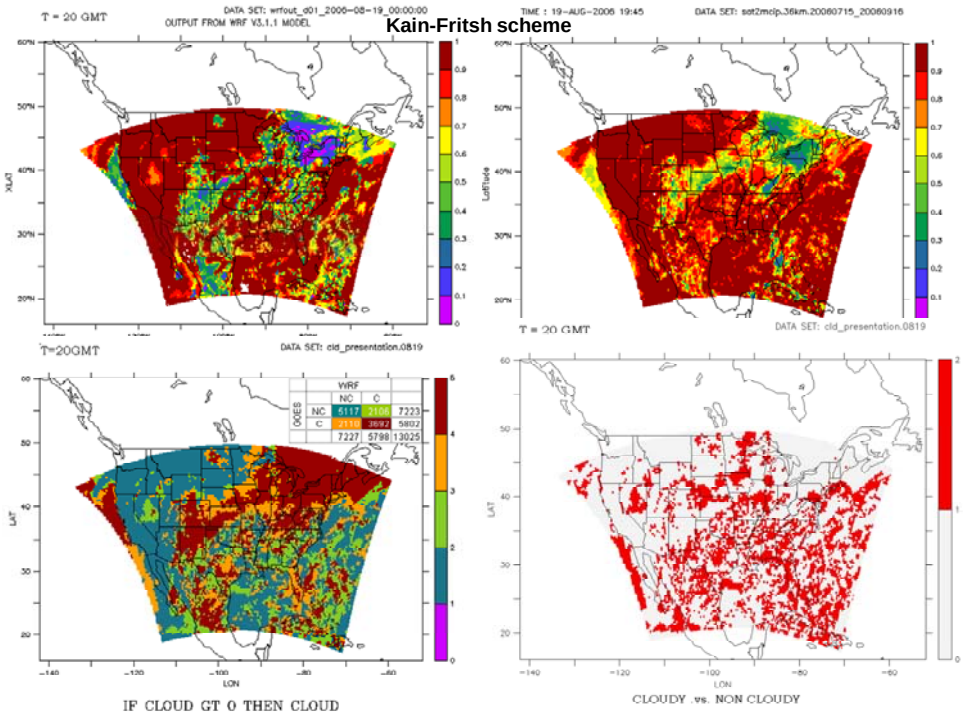


Figure 5.18 Comparison of model clouds versus satellite clouds. Upper left shows model clouds. Upper right shows GOES clouds. Lower left shows model cloud differences in a Cloud (C) and No Cloud (NC) categories graphically and in tabular staistical form (grid points).

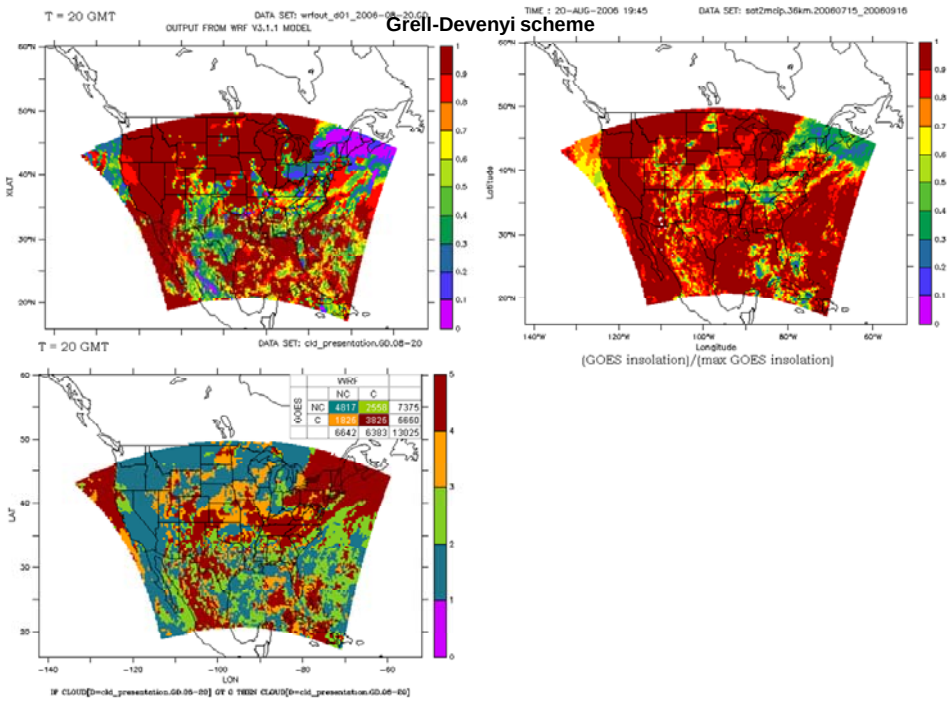


Figure 5.19 Same as figure 18 except for use of the Grell Devenyi convective parameterization.

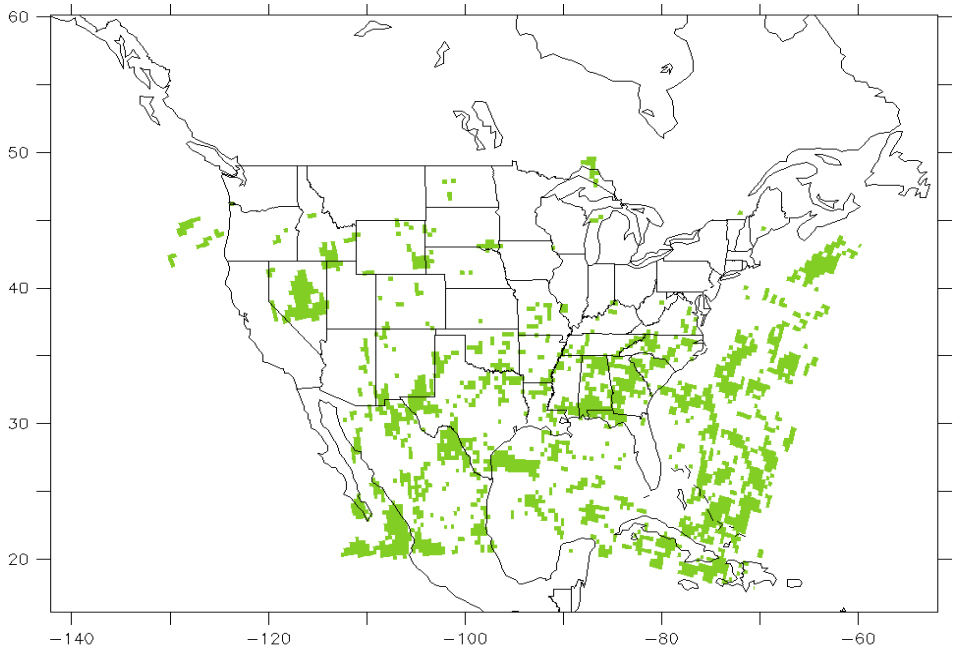


Figure 5.20 Image showing areas where model has clouds and GOES does not. This shows where clouds need to be suppressed.

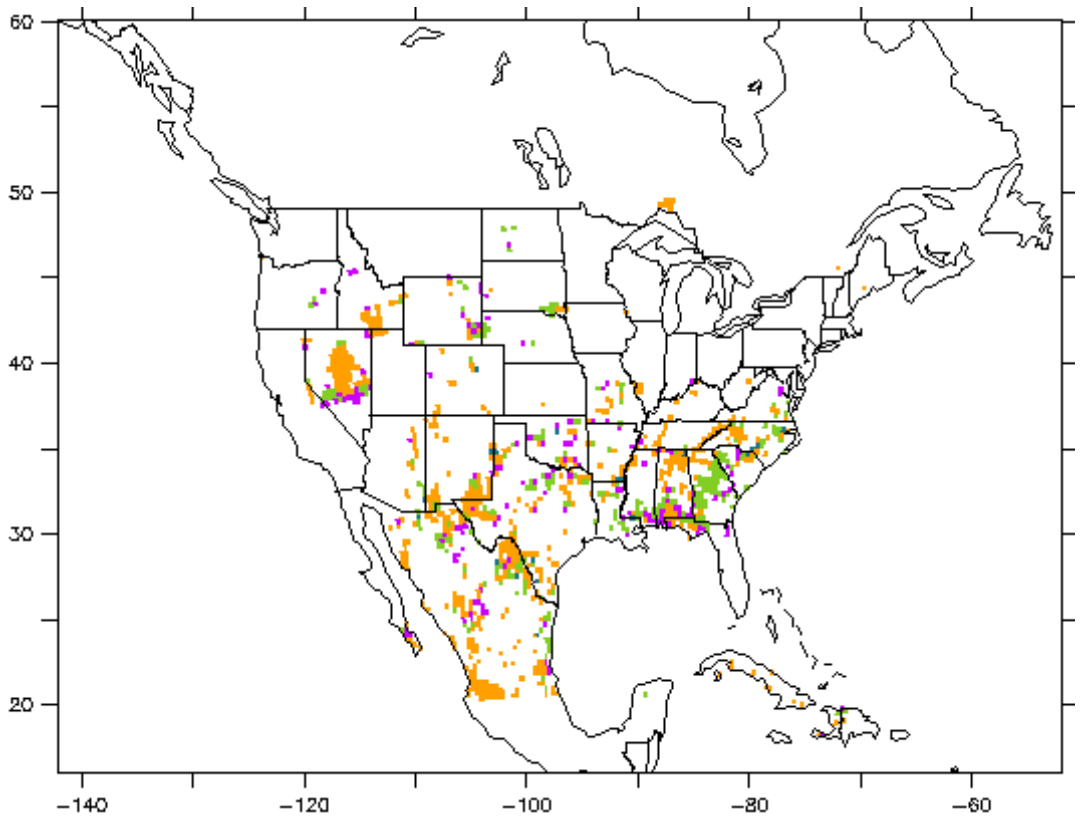


Figure 5.21 Depiction of where model has clouds and GOES does not. Yellow pixels indicate where model w>0.0. Green indicates where w<0.0. Thus, yellow indicates where clearing can perhaps be accomplished by imposing negative w.

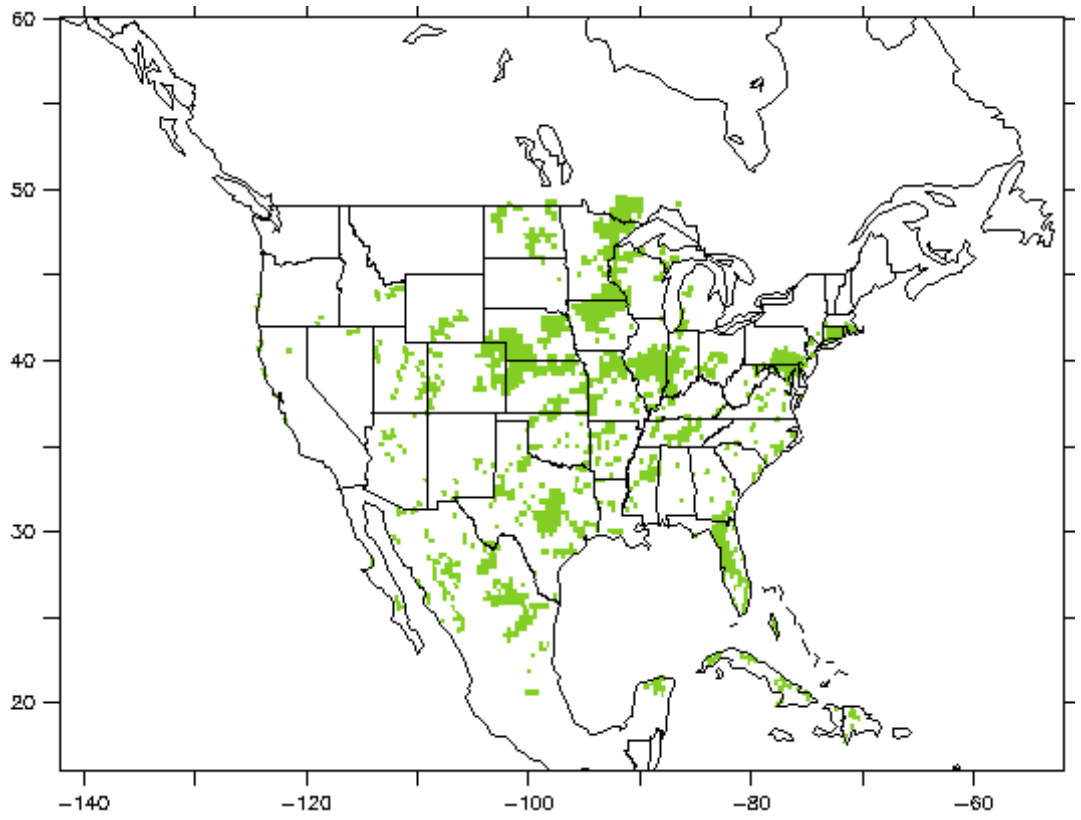


Figure 5.22 Depiction of where model is clear and GOES has clouds (green pixels). Show where clouds must be supported.

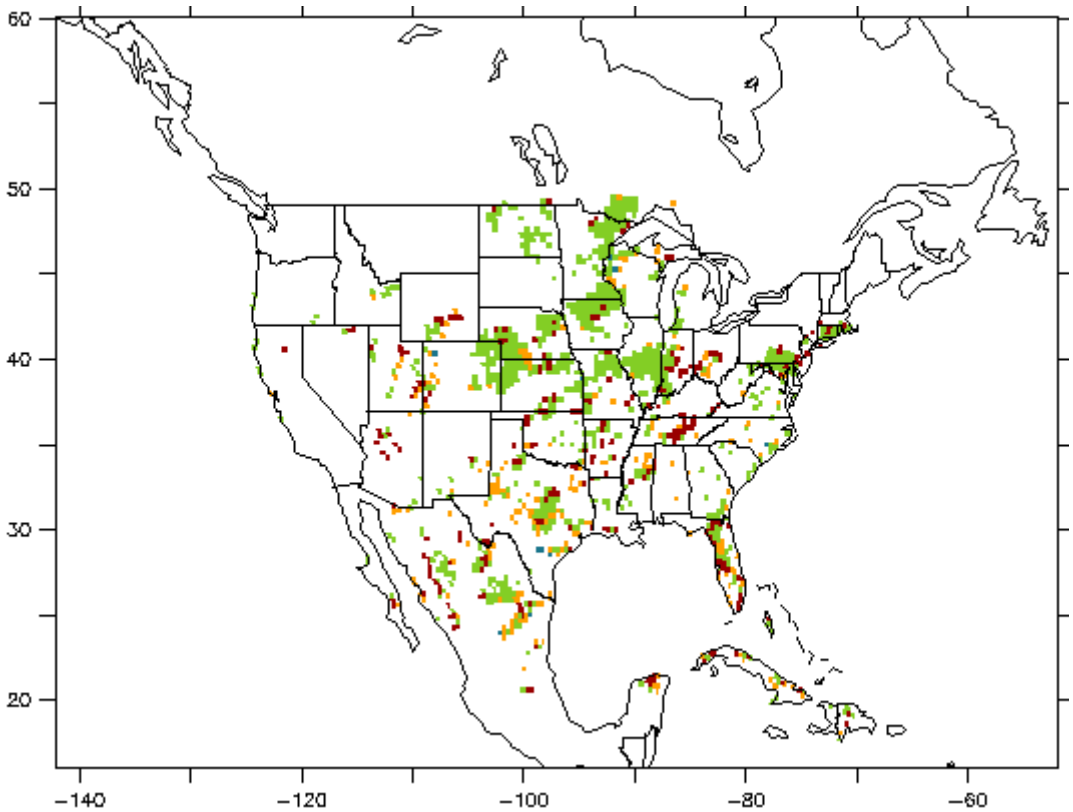


Figure 5.23 Depiction of where GOES has clouds but model has w<0.0. The green pixels show where the GOES has clouds but model w<0.0 . Thus, green pixels show where clouds can perhaps be created if W>0 can be imposed.

5.7 Monthly Statistics

The above depiction of cloud agreement between the model and GOES was based on discrete hours. The individual model events appear to support the hypothesis that clouds generally exists where the positive vertical velocities are present. Likewise clear areas are associated with negative vertical velocities. The results appear to indicate that it may be possible to clear clouds from the model where GOES does not have clouds by imposing a negative vertical motion or lower relative humidity. Likewise the results appear to show that in areas where the model does not have clouds and GOES does, that these areas have negative motion in the model. Thus, clouds might be supported by introducing positive vertical motion in the model and possibly higher humidity.

The relationships between model variable and clouds are now extended to monthly statistics. **Table 5.1** shows model and GOES statistics for the month of August 2006. Perhaps the strongest points in the table are that for WRF clouds (Case B) that .978 of clouds were associated with positive vertical motion and for clear conditions (Case C) that .665 of clear areas are associated with negative vertical velocities. Thus the table shows that clouds can perhaps be supported or suppressed if vertical velocities can be adjusted.

Table 5.1 Monthly average statistics for August 2006. Table shows fractions of grid points (pixels) satisfying the given contingency.

		WRF						
		Ncloud	Cloud	T = A+B+C+D				
	Ncloud	A	B	A,B,C,and D are distinguished by cloud albedo				
	Cloud	C	D	Cloud: cloud albedo>0.1				
				Ncloud: cloud albedo≤0.1				
						the fraction of clouds with w and rh		
for the case B				fraction of B (=B/T)		the fraction of clouds with w	rh>0.95	rh>0.9	rh>0.8
Overprediction				0.146	w>0.0	0.978 (=F1/E)	0.633(=G1/F1)	0.716 (=G2/F1)	0.851 (=G3/F1)
				fraction of clouds (=E/B)	w>0.02	0.825 (=F2/E)	0.603 (=H1/F2)	0.695 (=H2/F2)	0.843 (=H3/F2)
				0.768	w>0.05	0.465 (=F3/E)	0.606 (=I1/F3)	0.719 (=I2/F3)	0.877 (=I3/F3)
					w>0.10	0.18 (=F4/E)	0.641 (=J1/F4)	0.78 (=J2/F4)	0.937 (=J3/F4)
						fraction of clear sky in C (=K/C	fraction of clouds in C (=L/C)		
						0.906	0.094		
for the case C				fraction of C (=C/T)		the fraction of clear sky with w	the fraction of clouds with w	rh>0.8 in clear sky	rh>0.8 in clouds
Underprediction				0.14	w<0.0	0.655 (=N1/K)	0.345 (=M1/L)	0.704 (=P1/N1)	0.901 (=O1/M1
					w<0.02	0.665 (=N2/K)	0.335 (=M2/L)	0.703 (=P2/N2)	0.872 (=O2/M2
					w>0.0	0.345 (=N3/K)	0.655 (=M3/L)	0.726 (=P3/N3)	0.861 (=O3/M3
the number of grids in B with qcloud>10-5 (E)							the number of grids in C with qcloud≤10-5 (K)		
the number of grids in B with qcloud>10-5 and w>0.0 (F1)							the number of grids in C with qcloud>10-5 (L)		
the number of grids in B with qcloud>10-5 and w>0.02 (F2)							the number of grids in C with qcloud>10-5 and w<0.0 (M1)		
the number of grids in B with qcloud>10-5 and w>0.05 (F3)							the number of grids in C with qcloud>10-5 and w<0.02 (M2)		
the number of grids in B with qcloud>10-5 and w>0.10 (F4)							the number of grids in C with qcloud>10-5 and w>0.0 (M3)		
the number of grids in B with qcloud>10-5, w>0.0, and rh>95% (G1)							the number of grids in C with qcloud≤10-5and w<0.0 (N1)		
the number of grids in B with qcloud>10-5, w>0.0, and rh>90% (G2)							the number of grids in C with qcloud≤10-5 and w<0.02 (N2)		
the number of grids in B with qcloud>10-5, w>0.0, and rh>80% (G3)							the number of grids in C with qcloud≤10-5 and w>0.0 (N3)		
the number of grids in B with qcloud>10-5, w>0.02, and rh>95% (H1)							the number of grids in C with qcloud>10-5, w<0.0, and rh>80% (O1		
the number of grids in B with qcloud>10-5, w>0.02, and rh>90% (H2)							the number of grids in C with qcloud>10-5, w<0.02, and rh>80% (O		
the number of grids in B with qcloud>10-5, w>0.02, and rh>80% (H3)							the number of grids in C with qcloud>10-5,w>0.0, and rh>80% (O3)		
the number of grids in B with qcloud>10-5, w>0.05, and rh>95% (I1)							the number of grids in C with qcloud≤10-5, w<0.0, and rh>80% (P1)		
the number of grids in B with qcloud>10-5, w>0.05, and rh>90% (I2)							the number of grids in C with qcloud≤10-5, w<0.02, rh>80% (P2)		
the number of grids in B with qcloud>10-5, w>0.05, and rh>80% (I3)							the number of grids in C with qcloud≤10-5, w>0.0, and rh>80% (P3)		
the number of grids in B with qcloud>10-5, w>0.10, and rh>95% (J1)									
the number of grids in B with qcloud>10-5, w>0.10, and rh>90% (J2)									

5.8 Implementation in WRF

The examination of model statistical relationships between clouds and WRF model variables – vertical velocity and relative humidity – suggested a new alternate approach. Developing these relationships is comparable to developing the forward model Jacobian in 3D and 4D variational assimilation. While functional relationships were not clear, an examination of coincident

relations showed that threshold relations between vertical motion and relative humidity were very robust. For example for cloudy grid points in the model nearly 98% of the grids were associated with positive vertical motions. Also, for clear conditions over 65% of the grids were associated with negative vertical motions. This largely confirms the working hypothesis that in a GOES black and white image that white areas are associated with lifting and dark areas with subsidence.

The statistics also appear to show that in areas where the model has clouds but GOES does not that vertical motions are positive. Thus, if the vertical velocity can be adjusted to be negative then clouds can likely be cleared in the model. Likewise, it they show that where GOES has clouds and the model does not that clouds might be supported if vertical velocities can be adjusted to a positive value.

Thus the next logical step was to develop a methodology in which target vertical velocities are selected based on GOES images and the statistics developed here. The height of the target velocity will be based on observed GOES cloud tops. Using a one-dimensional variational approach, horizontal wind components in the model will be minimally adjusted (O'Brien 1970) to support the target vertical velocity. Coincidently, relative humidity will be adjusted below cloud to be consistent with the statistics presented here. This adjusted horizontal wind field and RH will then be nudged into the WRF or through nudging and/or a 3D Variational approach.

This methodology was put to test and a month-long simulation for the month of August was performed. The first step in the process was to define threshold vertical velocities based on model statistics. The results from control simulation were used to develop the needed statistics. **Table 5.2** shows the average vertical velocities from the model for cloudy and clear grid cells. The statistics were calculated separately over ocean and land as the air masses have different characteristics. These threshold velocities will be used as target vertical velocities needed to create or remove clouds in the model based on GOES observations. The table will be used as a look-up table to estimate target vertical velocities needed for cloud correction.

Table 5.2. Threshold Vertical Velocities Based on WRF Control Simulation.

		CLEAR				CLOUD			
		OCEAN		LAND		OCEAN		LAND	
height (m)		w (m/s)	RH (%)	w (m/s)	RH (%)	w (m/s)	RH (%)	w (m/s)	RH (%)
sfc	1000	-0.00253	72.08438	0.00377	39.52232	0.004865	99.12765	0.01269	99.6
1000	2000	-0.00588	59.14449	-0.00278	51.23995	0.034022	97.07111	0.02132	99.9
2000	4000	-0.00499	49.06997	-0.00745	41.42338	0.045954	95.62551	0.04551	100
4000	7000	-0.00608	40.36083	-0.01002	31.64465	0.054684	101.8438	0.06112	100
7000	10000	-0.01260	44.54638	-0.01433	36.94441	0.058007	99.79606	0.05639	98.96
10000	13000	-0.01579	47.13423	-0.01054	33.53775	0.065545	97.62615	0.05350	96.8
13000	~top	0.00018	33.25936	0.00067	19.85797	0.044565	94.18938	0.03255	93.2

In this first attempt in implementing the alternate approach cloud albedo was used to identify cloudy grid cells and therefore the assimilation was only performed during the day. The first step in this process is to identify the areas of over- and under-predictions. To do this GOES images are compared to model fields. **Figure 5.24** shows such comparison for August 19, 2006 at 19 GMT.

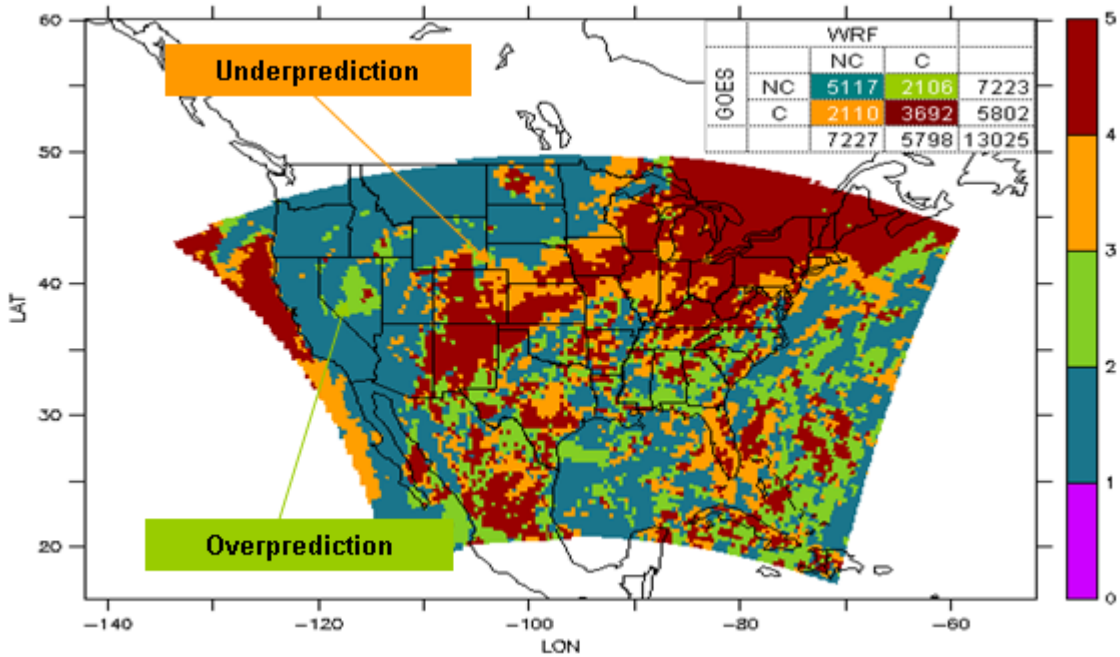


Figure 5.24. Image showing areas of agreement/disagreement between WRF and GOES (observed) cloud fields. Areas of agreement are depicted in blue (for clear sky) and red (for cloudy sky) while orange indicates under-prediction of clouds by WRF and green indicates over-prediction of clouds by WRF. The contingency table on the top right corner shows the number of grid cells in each category.

The second step in this process is to estimate a target vertical velocity. This is achieved by first identifying the cloud top height (or pressure) and cloud thickness. For the under-predicted grid cells, cloud top pressure and cloud albedo from GOES are used to estimate cloud top height and cloud thickness. For the grid cells where the model is over-predicting clouds, cloud water is used to map the cloud vertically. Then, using this information a target vertical velocity is estimated from Table 5.2.

Then, using a one dimensional variation technique (O'Brien 1970) horizontal wind components will be adjusted to sustain the target vertical velocities. The new horizontal wind field will be used as the nudging field and the model will be re-run for the past hour. Since GOES observations are hourly then the assimilation is also performed hourly.

This implementation is the first attempt at correcting the cloud fields with a much simpler technique. The main assumption in this first attempt is that the target (threshold) vertical velocities are adequate to lift a parcel to its saturation point or to move down a saturated air parcel to a point that it can no longer hold water and the relative humidity is substantially below 100%. Therefore, our expectation in this first attempt is to show marginal improvement in

model performance with respect to clouds. This is with the understanding that in this first attempt moisture (which is another key factor) is not being adjusted.

So, acknowledging these shortcomings the technique was applied to a month long simulation during August 2006. **Figure 5.25** shows a snapshot of wind fields before and after adjustment at 20 GMT on August 19, 2006. Color field contours indicate vertical velocity at 5km. The highlighted areas show the regions where the model has been under-predicting the clouds (center of domain, Oklahoma, Kansas, and north Texas) and regions where the model has been over-predicting clouds (eastern part of domain, over Atlantic off Carolina/Georgia coast). The figure on the right indicates that in both cases after adjustment the vertical velocities are reversed.

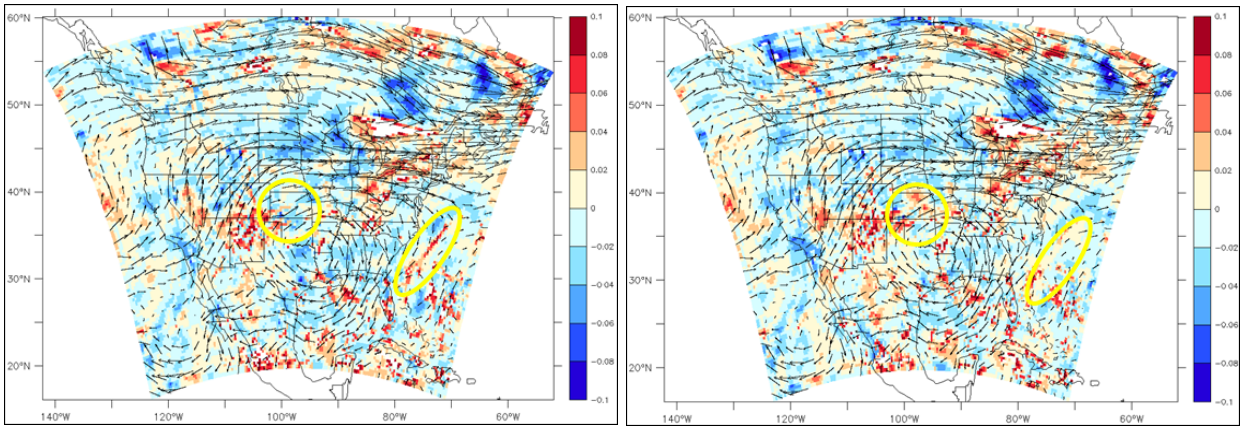


Figure 5.25. Snapshot of modified wind fields for 20 GMT, August 19, 2006. Color filled contours represent vertical velocity. Horizontal wind field was modified to sustain the target vertical velocity. The figure on the left is before the adjustment, while the figure on the right is after the adjustment. Highlighted areas show where subsidence is change to lifting and vice versa.

Several simulations were performed to test the effectiveness of the technique when configured with different convective parameterizations. As indicated in the previous sections, while the convective initiation in some convective parameterizations such as Kain-Fritsch (KF) are directly impacted by vertical velocity, the others, such as Grell scheme, are indirectly impacted.

Agreement Index (AI) was used as the metric for performance evaluation. **Figure 5.26** shows the AI for several simulations with and without cloud assimilation configured with New Grell, Grell-Devenyi, and Kain-Fritsch convective parameterizations. As evident from the figure, cloud assimilation has improved model performance regardless of the convective parameterization used. This is a very important conclusion as it shows that our technique is not dependent on the convective parameterization scheme and can be applied to different settings used in regulatory modeling practices. Also evident from the figure is the fact that the technique has more impact on model performance in some periods than others. For example, during the first two weeks of August when the synoptic forcing was stronger the improvements are more pronounced than the improvements on the third week of August when the synoptic forcing is weak. This could be due to the coarse grid spacing of 36 km used in these simulations that causes smaller average vertical velocities to be used as threshold. With vertical velocities not having the dynamic range needed for the convective events, larger scale cloudiness/clearing is better resolved (which is the case with stronger synoptic forcing) than initiating/suppressing convection needed for smaller scale

cloudiness/clearing. AI increase for the last week of August coincides with increased domain-wide cloudiness.

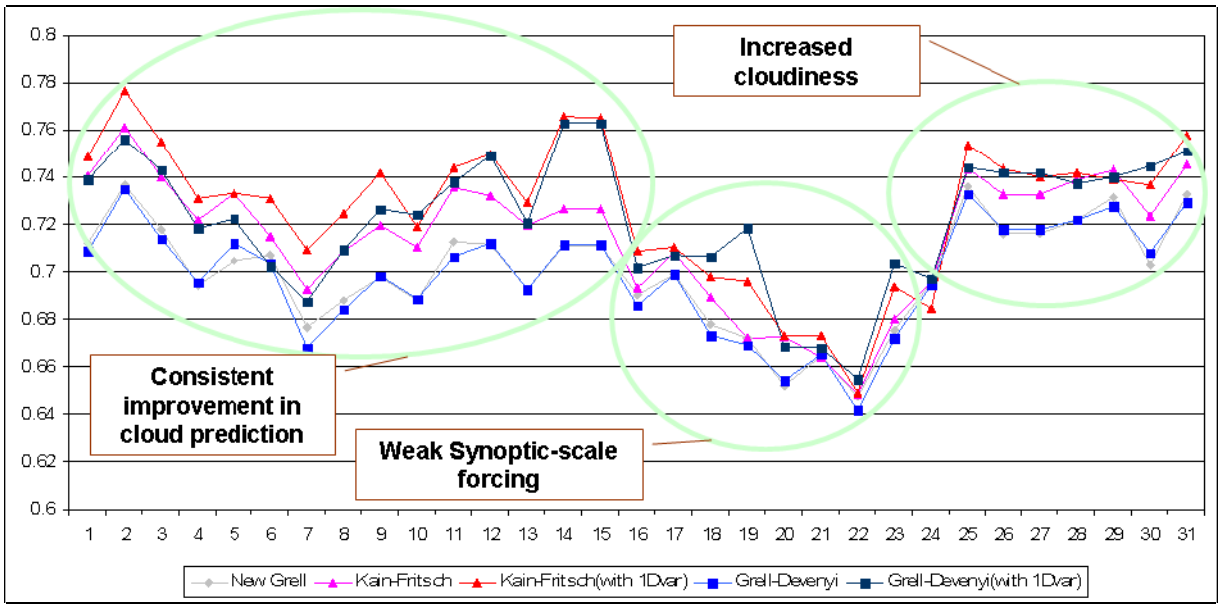


Figure 5.26. Agreement Index (AI) for month-long WRF simulations configured with New Grell, Grell-Devenyi, and Kain-Fritsch convective parameterizations. Also, in the figure are the results from two simulations with cloud assimilation configured with Kain-Fritsch and Grell-Devenyi.

While these results are preliminary, they are very encouraging. The results show that just adjusting the vertical velocity alone can make marginal improvement to cloud simulation regardless of the convective parameterization scheme used. Therefore, we can only expect better results when moisture adjustment is also applied in conjunction with the vertical velocity adjustment.

6 Summary and Conclusions

This report documented UAH's efforts in assimilating clouds in WRF using GOES observation. Two different paths were adopted to achieve the objectives of this project. The first path was to streamline the approach that was previously implemented and tested in MM5 and transition it to WRF modeling system, and the second path was to revisit the problem and devise a simplified approach that can be favorable to the operational regulatory applications.

An alternate simple approach was devised and partially implemented and tested for a month long simulation over August 2006. Only adjusting vertical velocities to dynamically support clouds where they are to be formed and remove them where there are erroneous model clouds proved to improve model performance regardless of the convective parameterization scheme used. However, in these first attempts moisture was not adjusted. The analysis of the success versus failure in cloud formation/removal indicated that moisture adjustment is necessary. This is due to the fact that this technique lifts an air parcel to its saturation point when it is necessary to create cloud and brings down an air parcel to a level where it is much below its saturation point when removing clouds. The technique fails to create clouds if the environment is excessively dry and saturation cannot be achieved.

The technique also relies on threshold vertical velocities that are obtained from model statistics. These statistics are dependent on grid-spacing used in a simulation and should be obtained from a relatively long control simulation. Our future work will address these issues by moving away from model statistics and analytically estimating vertical velocity and moisture adjustment needed to create/clear clouds.

The results in this study were based on a simulation with 36-km grid spacing which is relatively coarse relative to the observations. Since the resolution of GOES imager data is 4-km, the mapping of the data to 36-km model grid cell involves averaging the observation. This presents a problem when the grid is partially cloudy. One possibility is that a deep convective cell with high cloud albedo and low cloud top pressure that covers a portion of 36-km grid cell will be represented as a shallow cloud with lower cloud albedo and higher cloud top pressure that covers the entire 36-km grid cell. For this reason we envision that the model performance will improve even more as we approach the native resolution of GOES data in the simulation. This needs to be tested in future works.

References

Albers, S. C., J. A. McGinley, D.L. Birkenheier, and J.R. Smart, 1996: The local analysis and prediction system (LAPS): Analyses of clouds, precipitation, and temperature. *Wea. Forecasting*, **11**, 273-287.

Diak, G. R. and C. Gautier, 1983: Improvements to a simple physical model for estimating insolation from GOES data. *J. Appl. Meteor.*, **22**, 505–508.

Lipton, A.E., and G.D. Modica, 1999: Assimilation of Visible-Band Satellite Data for Mesoscale Forecasting in Cloudy Conditions. *Mon. Wea. Rev.*, **127,** 265-278.

Lopez, P., 2007: Cloud and Precipitation Parameterizations in Modeling and Variational Data Assimilation: A Review. *J. Atmos. Sci.*, **64**, 3766–3784.

McNider, R. T., A. J. Song, D. M. Casey, P. J. Wetzel, W. L. Crosson, and R. M. Rabin, 1994: Toward a dynamic-thermodynamic assimilation of satellite surface temperature in numerical atmospheric models. *Mon. Wea. Rev.*, **122**, 2784–2803.

McNider, R.T., A. Song, and S.Q. Kidder, 1995a: Assimilation of GOES-derived solar insolation into a Mesoscale model for studies of cloud shading effects. *Int. J. Remote Sens.,* **16,** 2207-2231.

McNider, R. T., X. Shi, M. Friedman, and D. E. England,1995b: On the predictability of the stable atmospheric boundary layer, *J. Atmos. Sci.,* **52**, 1602-1614.

Mu, M., and J. Wang, 2003: A Method for Adjoint Variational Data Assimilation with Physical "On–Off" Processes. *J. Atmos. Sci.*, **60**, 2010–2018.

Pour-Biazar, Arastoo, Richard T. McNider, Shawn J. Roselle, Ron Suggs, Gary Jedlovec, Soontae Kim , Daewon W. Byun, Jerry C. Lin, Thomas C. Ho, Stephanie Haines, Bright Dornblaser, Robert Cameron. (2007), Correcting photolysis rates on the basis of satellite observed clouds, *J. Geophys. Res*, 112, D10302, doi:10.1029/2006JD007422.

Stephens, G.L., 1978: Radiation profiles in extended water clouds. Part II. Model Parameterizations. *J.Atmos. Sci*., 35, 2123-2132

Stephens, Graeme L., Deborah G. Vane, Ronald J. Boain, Gerald G. Mace, Kenneth Sassen, Zhien Wang, Anthony J. Illingworth, Ewan J. O'Connor, William B. Rossow, Stephen L. Durden, Steven D. Miller, Richard T. Austin, Angela Benedetti, Cristian Mitrescu, and CloudSat Science Team, 2002: The CloudSat mission and the A-train: A new dimension of space-based observations of clouds and precipitation. *Bull. Amer. Meteorol. Soc.*, **83**, 1771-1790, doi:10.1175/BAMS-83-12-1771

Yucel, I., W. J. Shuttleworth, X. Gao, and S. Sorooshian, 2003: Short-term performance of MM5 with cloud-cover assimilation from satellite observations. *Mon. Wea. Rev.*, **131**, 1797–1810.

APPENDIX A

Convective Suppression
Detailed Description

1 Introduction to Convective Suppression

The following documents the implementation of the convective suppression strategy. In addition to the convective suppression path, considerable progress was made in developing relationships in the model for adjusting vertical velocity and relative humidity.

This activity was part of task 2 in the proposal work plan: "Evaluation of model performance including only a suppression of the convective parameterizations based on satellite data". This work used the Weather Research and Forecasting (WRF) Model and more specifically Version 3.1.1 of the Advanced Research WRF (ARW) and the related WRF Preprocessing System (WPS). Section 2 describes the GOES data used in this task, section 3 gives an overview of the WPS and WRF changes, section 4 shows how the GOES data is involved in the input/output (IO) in WPS and WRF, section 5 discusses the required steps to implement the use of the GOES data, section 6 shows the results of a short test run, section 7 provides references.

The test run was a 30-km 72-h WRF simulation from 00 UTC 4 February through 00 UTC 7 February of 2008 using the Kain-Fritsch cumulus parameterizationn. This short case was chosen as an environment for creating and testing the GOES data implementation prior to using it on the June – September 2006 simulation. Three simulations were performed with the test case: 1) a single 30-km grid, 2) 30-km and 10-km grids in a two-way nest mode, and 3) 30-km and 10-km grids in a one-way nest mode have been run and examined. Only the first simluation will be discussed in this report.

2 GOES Data

Table 1 illustrates the variables, original resolution, and units of the satellite data used in this software which are available for two types of GOES instruments – Imager (IMG) and Sounder (SND). Both the skin temperature and cloud top pressure retrievals rely on NAM model first-guess data. Cloud top pressure (CTP) is available as an Imager or Sounder product. The Imager product is approximately 4 km resolution, but is a simple infrared temperature to pressure retrieval. The Sounder product is at a coarser resolution of about 10 km, but is a combined retrieval method using the IR temperature for lower clouds and CO_2 slicing method for the higher clouds and is therefore a more robust CTP product. The column "Primary or Derived" in Table 1 indicates whether a variable is directly available (i.e. a primary variable) or is derived one. The field used to determine cloud cover percentage in this report is "CSKYPGOES" and was based on the percentage of an analysis grid cell covered by clouds based on the Imager cloud top pressure field (CTOPPGOES). The other secondary variable, the clear-sky percentage (CLEARGOES), was based on the percentage of an analysis grid cell covered by clouds based on the Imager cloud albedo field (CTOPPGOES). "CSKYPGOES" was chosen at this stage of the research primarily because it is available day and night as opposed to only a daytime product like CLEARGOES. The nocturnal cloud field which remains until morning can have a strong modulating influence on the daytime convection. Areas which are cloudy in the morning can delay or prevent daytime convection for summertime cases where the synoptic forcing is weak. For this reason and others, control of the nighttime cloud field

is important. Although it probably is a better product, the Sounder cloud top pressure is not currently being used in this research application because in its present form it is available only for the Eastern U.S. This report presupposes familiarity and use of the website http://satdas.nsstc.nasa.gov/, which provides access to the GOES data, the documentation, and software to regrid the GOES data to the WRF model nested grids.

Table 2.1 Description of GOES data used in WPS/WRF package

DATA MODE	VARIABLE	ORIGINAL RESOLUTION	NetCDF VARIABLE NAME	UNITS	Primary Or Derived
IMG	Cloud Albedo	4 km	CDALBGOES	%	Primary
IMG	Cloud Top Pressure	4 km	CTOPPGOES	mb	Primary
IMG	Insolation	4 km	INSOLGOES	Watts m^{-2}	Primary
IMG	Surface Albedo	4 km	SFALBGOES	%	Primary
SND	Cloud Top Pressure	10 km	not used	mb	Primary
SND	Skin Temperature	10 km	TSKINGOES	K	Primary
IMG	Cloud Percentage	4-km	CSKYPGOES	%	Derived
IMG	Clear Sky Percentage	4-km	CLEARGOES	%	Derived

3 Overview of Changes

There were two goals in mind as the system changes were developed to implement use of the GOES data in the WPS/WRF system to suppress convection where the GOES data indicated clear conditions. The first goal was to make changes in as few files as possible, and the second was to utilize the existing WRF input/output (IO) system. Apart from developing the necessary IO changes to ingest the GOES data, the main challenge was introducing time interpolation into the WPS program **metgrid.exe**. The so-called "first-guess" (FG) meteorological data sources typically have a time frequency of 3-6 h. The current GOES data frequency is about one hour, with products having a timestamp of about 45 minutes past the hour. It was therefore necessary to perform linear interpolation in time for the FG fields, and linear interpolation in time or the nearest in time (depending on the variable involved) for the GOES data to create hourly output fields from **metgrid.exe**. If in the future higher frequency satellite data are available, only simple changes in the **namelist.wps** and **namelist.input** files would be needed. Section 3.1 gives an overview of the WPS modifications, and section 3.2 gives the same for the WRF package. Section 3.3 describes the changes in the GOES regridding software.

3.1 WPS Modifications

The **namelist.wps** file is the control file for the WPS programs **geogrid.exe**, **ungrib.exe**, and **metgrid.exe.** Table 3.1 shows an example of the additions needed to the **namelist.wps** file in order to utilize the GOES data. The additions are color coded in Table 3.1 to clarify the changes. The red row shows the definition of a new namelist group called "goes_processing". To enable the processing of the GOES data the variable "process_goes_data" (blue row) must be set to ".true.". The variable "goes_interval_seconds" (green row) defines the approximate frequency in seconds of the

GOES data – here set at 3600 s. The pink rows are one or more full directory paths and file prefixes of the processed ASCII GOES data. There should be one path and file prefix for each nested grid to be considered in the WPS/WRS package. The last set of variables (yellow rows) define the variable name, the time analysis method, and time constraints in minutes for the specified method. The latter three fields can be separated by spaces and/or tabs. The variable names are set in the **Registry.EM** file (discussed in section 3.2) and are the names which then appear in the NetCDF files produced by the programs **metgrid.exe**, **real.exe**, and **wrf.exe**. Two options exist for the time analysis method – "time_near" and "time interpolation". The option "time_near" assigns the nearest GOES data to the analysis time as long as it is no further away than the specified time constraint. The option "time_interp" performs linear interpolation in time from the available GOES data to the analysis times as long as the data do not violate the specified time constraint. For the typical user, only the "goes_name" variable will need to be changed.

Table 3.1 Example of additions to the **namelist.wps** file to process GOES data.

```
&goes_processing
 process_goes_data = .true.
 goes_interval_seconds = 3600
 goes_name = '/rstor/dotykg/GOES_APP_FEB_2010/DATA_ASCII_OUT-30km/g8i_rtv_30km',
             '/rstor/dotykg/GOES_APP_FEB_2010/DATA_ASCII_OUT-10km/g8i_rtv_10km'
 goes_options_in = 'CSKYPGOES time_near      30',
                   'CDALBGOES time_near      30',
                   'TSKINGOES time_interp    60',
                   'SFALBGOES time_interp    60',
                   'INSOLGOES time_interp    60',
                   'CLEARGOES time_near      30',
                   'CTOPPGOES time_near      30',
/
```

The discussion of the F90 code changes implemented in the WPS follow the information in Table 3.2. The four F90 source files which were changed were: **process_domain_module.F**, **metgrid.F**, **process_domain_module.F**, and **read_met_module.F**. Most of the changes were confined to the main program driver file **metgrid.F**. These changes are automatically done by the script **goes_changes.csh** descibed in section 5.

Thirteen new subroutines were added to **metgrid.F**. Five of these were related to date calculations (mainly calculating an epochal time in minutes given a year, month, day, and UTC time in hours and minutes; and then the inverse of the latter operation). An extensive Earth System Modeling Framework (ESMF) date calculation package is part of WRF, but documentation was non-existent and no usage in WRF could be located which illustrated how to perform the needed calculations. In the future the ESMF date functions will likely replace the date subroutines included in this revision package. The remaining new subroutines in **metgrid.F** will now be briefly discussed. Subroutine **get_fg_file_times** is the main driver for processing the GOES data and calculates the data file weighting as a function of the first-guess data type, variable, analysis time, and grid. Subroutine **get_logical_unit** chooses an unused logical unit for reading or writing. Subroutine **write_goes_intermediate_data** writes out the GOES data into the WPS binary (FORTRAN unformatted write) intermediate format. Subroutine **get_list_of_goes_files** creates lists of the ASCII GOES files defined by the namelist

variable "goes_name" illustrated in Table 3.1. Subroutine **parse_goes_input_options** parses the three fields of the namelist variable "goes_options_in" shown in Table 3.1. Subroutine **get_dx_from_fname** extracts the horizontal resolution in km from the ASCII GOES filenames. Subroutine r**ead_goes_record** is used to read the ASCII GOES data files. The final new subroutine in file **metgrid.F** is **findilev** which locates the two array elements which bracket a search value in a sorted list of values in an integer array.

The remaining changes in the WPS system were modifications to existing subroutines. New variable definitions were declared in the file **gridinfo_module.F**. Changes to the file **process_domain_module.F** enabled the processing of the GOES data and the use of the time analysis options of "time_near" and "time_interp". Subroutine r**ead_met_init** in file **read_met_module.F** was modified to enable filename creation by time (the original and only way) and by a passed argument.

The final set of required changes in WPS comprise additions to the **METGRID.TBL** file as given in Table 3.3. An example file **METGRID.TBL.GOES** with these additions is part of the tar file described in section 5 and the script **goes_changes.csh** by default replaces the existing **./metgrid/METGRID.TBL** file with the latter file. The **METGRID.TBL** files which come with WRF version 3.1.1 distribution include the following: **METGRID.TBL.AFWA**, **METGRID.TBL.ARW**, **METGRID.TBL.ARW.ruc**, and **METGRID.TBL.NMM**. If the **METGRID.TBL.GOES** file is not appropriate, then additional manual steps would be needed after running the script **goes_changes.csh**. As an example, if the file **METGRID.TBL.ARW** is the appropriate choice, then steps similar to the following would then be required in the directory **./WPS/metgrid**:

1. cp **METGRID.TBL.ARW METGRID.TBL.ARW.GOES**

2. Append changes as described in Table 3.3 in the file **METGRID.TBL.GOES** to file **METGRID.TBL.ARW.GOES**

3. ln –s **METGRID.TBL.ARW.GOES METGRID.TBL**

Table 3.2 Summary of WPS code changes to process GOES data. Subroutine names in orange are new, whereas subroutine names in purple are original WPS code which was modified.

FILE	Subroutine Name	Comment
/WPS/metgrid/src/gridinfo_module.F		Added variable definitions
/WPS/metgrid/src/gridinfo_module.F	**get_namelist_params**	Added namelist "goes_processing"
./WPS/metgrid/src/metgrid.F	**get_fg_file_times**	Calculates file weighting as a function of first-guess data type, variable, analysis time, and grid
./WPS/metgrid/src/metgrid.F	**get_logical_unit**	Chooses an unused logical unit for IO
./WPS/metgrid/src/metgrid.F	**calc_julian_minute**	Calculates an epochal time in minutes since 00 UTC 1 Jan 1990 given the year, month, day, and UTC time
./WPS/metgrid/src/metgrid.F	**getjtime**	Calculates julian time in minutes relative to beginning of year given year, month, day, hour, and minute
./WPS/metgrid/src/metgrid.F	**inverse_julian**	Given year and julian day, calculate month and day
./WPS/metgrid/src/metgrid.F	**inverse_getjtime**	Given a julian time, calculate a julian day and the UTC time in hours and minutes
./WPS/metgrid/src/metgrid.F	**inv_julian_minute**	Inverse of calc_julian_minute
./WPS/metgrid/src/metgrid.F	**write_goes_intermediate_data**	Writes GOES data out in WPS binary intermediate format
./WPS/metgrid/src/metgrid.F	**get_list_of_goes_files**	Create list of GOES ASCII files
./WPS/metgrid/src/metgrid.F	**parse_goes_input_options**	Parse goes-processing information from namelist.input
./WPS/metgrid/src/metgrid.F	**get_dx_from_fname**	Extract grid resolution in km from GOES ASCII file name
./WPS/metgrid/src/metgrid.F	**read_goes_record**	Read GOES ASCII files
./WPS/metgrid/src/metgrid.F	**findilev**	search a sorted integer array for a specific value
./WPS/metgrid/src/process_domain_module.F	**process_domain**	Changes to enable processing of GOES data
./WPS/metgrid/src/process_domain_module.F	**process_single_met_time**	Changes to enable processing of GOES data and time interpolation
./WPS/metgrid/src/read_met_module.F	**read_met_init**	Create file name by time or passed argument

Table 3.3 Additions to METGRID.TBL file to process GOES data

==========	==========
name=CSKYPGOES	**name=INSOLGOES**
output=yes	**output=yes**
mandatory=yes	**mandatory=yes**
output_stagger=M	**output_stagger=M**
interp_option=nearest_neighbor	**interp_option=nearest_neighbor**
fill_missing= -9999.0	**fill_missing= -9999.0**
missing_value= -999.0	**missing_value= -999.0**
==========	==========
name=CDALBGOES	**name=CLEARGOES**
output=yes	**output=yes**
mandatory=yes	**mandatory=yes**
output_stagger=M	**output_stagger=M**
interp_option=nearest_neighbor	**interp_option=nearest_neighbor**
fill_missing= -9999.0	**fill_missing= -9999.0**
missing_value= -999.0	**missing_value= -999.0**
==========	==========
name=TSKINGOES	**name=CTOPPGOES**
output=yes	**output=yes**
mandatory=yes	**mandatory=yes**
output_stagger=M	**output_stagger=M**
interp_option=nearest_neighbor	**interp_option=nearest_neighbor**
fill_missing= -9999.0	**fill_missing= -9999.0**
missing_value= -999.0	**missing_value= -999.0**
==========	==========
name=SFALBGOES	
output=yes	
mandatory=yes	
output_stagger=M	
interp_option=nearest_neighbor	
fill_missing= -9999.0	
missing_value= -999.0	
==========	

3.2 WRF Modifications

The **namelist.input** file is the control file for the programs **real.exe** and **wrf.exe.** Table 3.4 shows an example of the additions needed to the **namelist.input** file in order to utilize the GOES data. The additions are color coded to clarify the changes. The yellow section indicates changes which should be added to the end of the existing namelist group "time_control" section. These specify the name and time frequency for the "met_em*nc" files produced by program **metgrid.exe** and are identified for input on IO stream number 6. The red row shows the definition of a new namelist group called "goes_data". The namelist variable "do_conv_off" (blue row) must be set to either ".true." or ".false." for each nested grid. When set to ".true.", the convective parameterization specified for that grid is not called when successive hours are "clear" as defined by the second namelist variable "cloud_threshold" (green row). As defined in the example in Table 3.4, when cloud coverage is 33% or less the grid cell is deemed "clear".

The discussion of the F90 code changes implemented in WRF follow the information in Table 3.5. The five F90 source files which were changed were: **module_first_rk_step_part1.F**, **real_em.F**, **module_cu_kfeta.F**, **input_wrf.F**, and **mediation_integrate.F**. The modifications can be summarized as making IO changes to

read the GOES data and adding subroutine arguments to pass the needed variables to the correct subroutines. These changes are automatically done by the script **goes_changes.csh** descibed in section 5.

The final set of required changes in WRF package comprise additions to the **Registry.EM** file as given in Table 3.6. The four types of changes have been color coded to clarify the modifications. The first group of changes (pink rows) define the GOES variables and designate them for input on IO stream number six and output to the standard history. The second group of changes (yellow rows) add the namelist group "goes_data" and define the input options "do_conv_off" and "cloud_threshold", which are set in the file **namelist.input** as described above. An example **Registry.EM.GOES** file with these changes is included in the tar file **tar.goes.changes** described in section 5. The script **goes.changes.csh** by default replaces the current **Registry.EM** file with the **Registry.EM.GOES** file. If the user desires something different, then the changes described in Table 3.6 will have to be manually entered into the user's choice for the **Registry.EM** file.

Table 3.4 Example of changes to the **namelist.input** file to use GOES data.

&time_control
auxinput6_inname = "met_em.d<domain>.<date>.nc"
auxinput6_interval_h = 01
&goes_data
do_conv_off = .true., .true.,
cloud_threshold = 33.0, 33.0,
/

3.3 Changes to the GOES Regridding Software

The script **goes.changes.csh** automatically updates the GOES regridding "REFORMAT/reformat.f90"file and recompiles the same program. The user should not have to make any changes to this update. This change adds the analyzed cloud cover percentage based on the Imager cloud top pressure field to the ASCII output data.

Table 3.5 Summary of WRF changes to implement use of GOES data. Subroutine names in purple designate existing subroutines which were modified.

FILE	Subroutine Name	Comment
./WRFV3/dyn_em/module_first_rk_step_part1.F	first_rk_step_part1	Add arguments to "call cumulus_driver" statement
./WRFV3/main/real_em.F	assemble_output	Make changes to insert GOES cloud percentage into fdda2d arrays
./WRFV3/phys/module_cu_kfeta.F	KF_eta_CPS	Add arguments and changes to use GOES data cloud percentage
.WRFV3/phys/module_cu_kfeta.F	KF_eta_PARA	Add arguments and changes to use GOES data cloud percentage
./WRFV3/share/input_wrf.F	input_wrf	Make changes for IO stream number 6
./WRFV3/share/mediation_integrate.F	med_before_solve_io	Dedicate IO stream 6 for GOES data

Table 3.6 Summary of changes to Registry.EM file to enable GOES data processing.

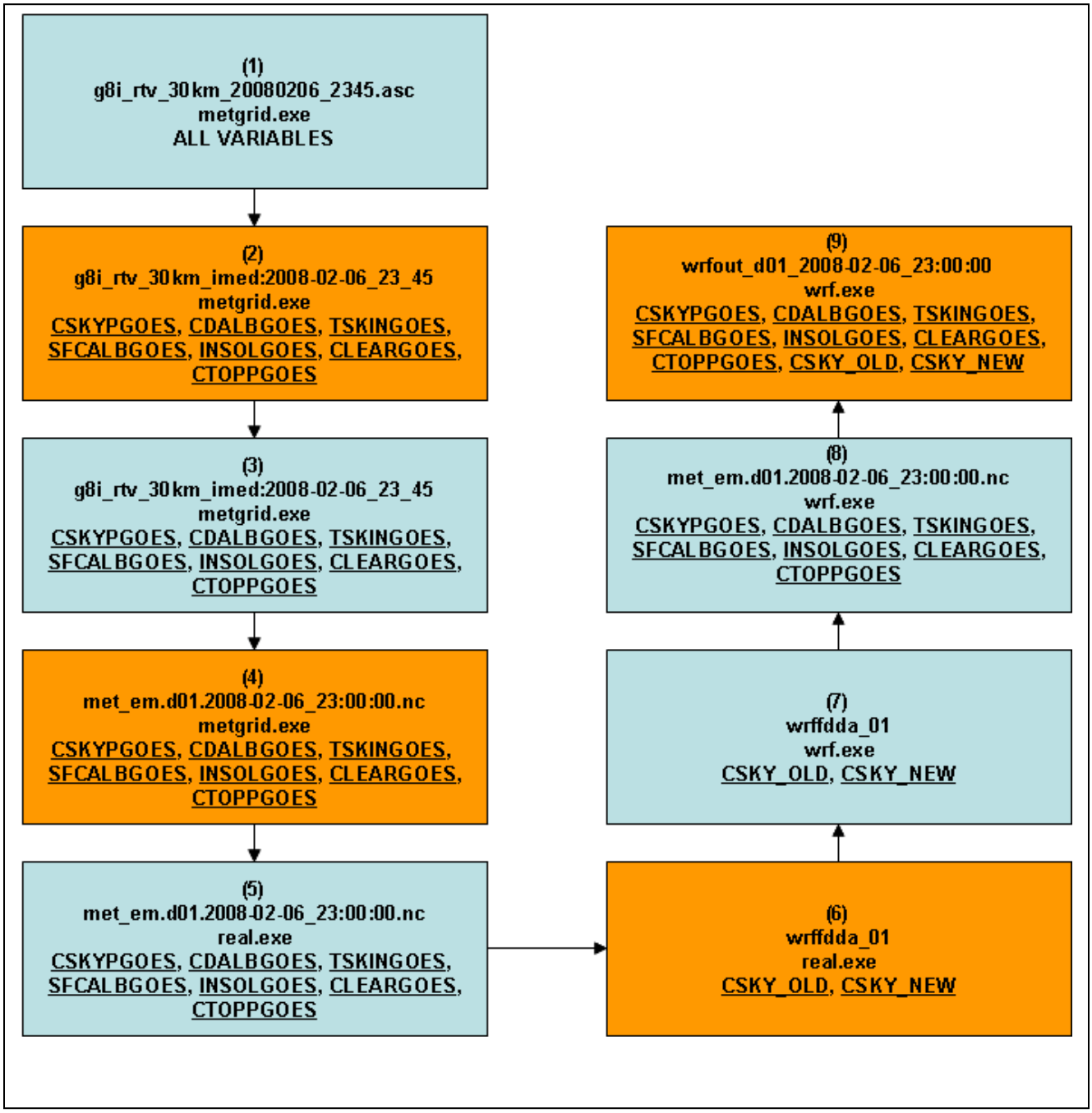
`state real cskypgoes ij dyn_em 1 - i16h "CSKYPGOES" "cloud cover percentage 0-100" "%"`
`state real cdalbgoes ij dyn_em 1 - i16h "CDALBGOES" "cloud albedo percentage 0-100" "%"`
`state real tskingoes ij dyn_em 1 - i16h "TSKINGOES" "skin temperature" "K"`
state real sfalbgoes ij dyn_em 1 - i16h "SFALBGOES" "surface albedo percentage 0-100" "%"
`state real insolgoes ij dyn_em 1 - i16h "INSOLGOES" "GOES insolation" "Watts m-2"`
state real cleargoes ij dyn_em 1 - i16h "CLEARGOES" "clear sky percentage 0-100" "%"
`state real ctoppgoes ij dyn_em 1 - i16h "CTOPPGOES" "cloud top pressure" "mb"`
`rconfig logical do_conv_off namelist,goes_data max_domains .false. irh "do_conv_off" "T/F PREVENT CONVECTIVE PARAMETERIZATION ACTIVATION FOR CLEAR SKIES"`
`rconfig real cloud_threshold namelist,goes_data max_domains 33.0 irh "cloud_threshold" "0.0-100.0 CLOUD FRACTION THRESHOLD < this value: clear, > this value: cloudy"`
`package kfetascheme cu_physics==1 - fdda2d:csky_old,csky_new`

4 Data Flow

This discussion of the data flow of the GOES information is based on the information in Figure 4.1. Two related but different IO streams use the GOES data. One IO stream adds the cloud cover percentage (variable CSKYPGOES) to the two-dimensional grid analysis nudging arrays in **real.exe** which results in the new variables “CSKY_OLD” and “CSKY_NEW” being output to the **wrffdda** files. These new variables are then available within **wrf.exe** and are output to the WRF history files. The latter two variables give the hourly values which encompass the model integration steps between the two hourly values. This approach was taken because WRF was already set up to read two time values of the specified grid analysis ffda variables set in the **Registry.EM** file (see section 3.2). This requires that analysis nudging be turned “on” for all the nested grids where the GOES data are to be used. If analysis nudging is not desired for a given grid where GOES data are needed, the nudging weights can be set to zero for that grid. The second IO stream adds all the hourly specified GOES variables (see Table 2.1) from the met_em*nc files output from **metgrid.exe** to the WRF output history files. This is accomplished by changes to the **Registry.EM** file and the **namelist.input** file as described in section 3.2.

The changes made to the WPS program **metgrid.exe** enable the ASCII GOES data to be read and then output in the WPS intermediate format. The existing WPS code is then able to read the GOES data in this intermediate format.

Figure 4.1 Data flow of GOES data from the program **metgrid.exe** to **wrf.exe**. Blue boxes designate read operations, and orange boxes denote write operations. The first line in each box is a sequential step number. The second line is an example of the filename in the read or write operation. The third line is the program which performs the IO function. Lines four and beyond give the variable names involved.



5 Implementation

The tar file **tar.goes.changes** contains all the modifications or examples thereof for implementation of the GOES processing changes. A listing of the files in **tar.goes.changes** is given in Table 5.1. All the F90 files are automatically updated by

Table 5.1 Listing of files from tar file **tar.goes.changes**.

goes_changes.csh
gridinfo_module.F
input_wrf.F
mediation_integrate.F
metgrid.F
METGRID.TBL.GOES
module_cu_kfeta.F
module_cumulus_driver.F
module_first_rk_step_part1.F
namelist.input.GOES
namelist.wps.GOES
process_domain_module.F
read_met_module.F
real_em.F
reformat.f90
Registry.EM.GOES

the script **goes_changes.csh**. Depending on the circumstance, manual editing of the files **METGRID.TBL**, **namelist.wps**, **namelist.input**, **Registry.EM**, and the script (user provided) which runs **metgrid.exe** may be needed. A simplified list of steps is given in section 5.1, and a more detailed discussion is given in section 5.2.

5.1 Simple Description of Implementation Steps

The following steps assume the user has downloaded the file **tar.goes.changes** to a user defined directory (here called "WUPDATE") and that the latter directory is the current directory. The step numbers correspond to those in Figure 5.1.

3. tar –xvf **tar.goes.changes**
4. edit directory paths in script **goes.changes.csh**
5. **./goes_changes.csh**
6. process or reprocess GOES data
9. edit/replace **METGRID.TBL**, **namelist.input, namelist.wps** and **Registry.EM** as needed
13. add "rm DONE" to script (provided by user) that runs **metgrid.exe**

16, 17. link met_em*nc files from **metgrid.exe** to directories where **real.exe** and **wrf.exe** are to be run

5.2 Discussion of Implementation Steps

A flow chart summarizing the main steps in implementing the GOES data are illustrated by Figure 5.1. Blue boxes denote standard steps which are needed regardless of whether GOES data are being used. Orange boxes denote steps required to implement the

processing of the GOES data and only these will be discussed in this section. Each action box has a step number and these will be used in the following discussion:

Step 2. Involves acquiring the file **tar.wrf.goes.changes** and placing it in a user specified directory (called "WUPDATE" in Figure 5.1).

Step 3. tar –xvf **tar.goes.changes**

Step 4. Edit the full directory paths at the top of the script **goes_changes.csh**

Step 5. **./goes_changes.csh**

This step copies the needed files to a backup unique name in the same directory and replaces the same files with new versions containing the changes described in section 3 . For example, the file **metgrid.F** would be copied to a file with a pattern like **metgrid.F.KEEP.Jun.30.2010.095140**, where the last six digits are the two-digit hour, minute, and second, respectively.

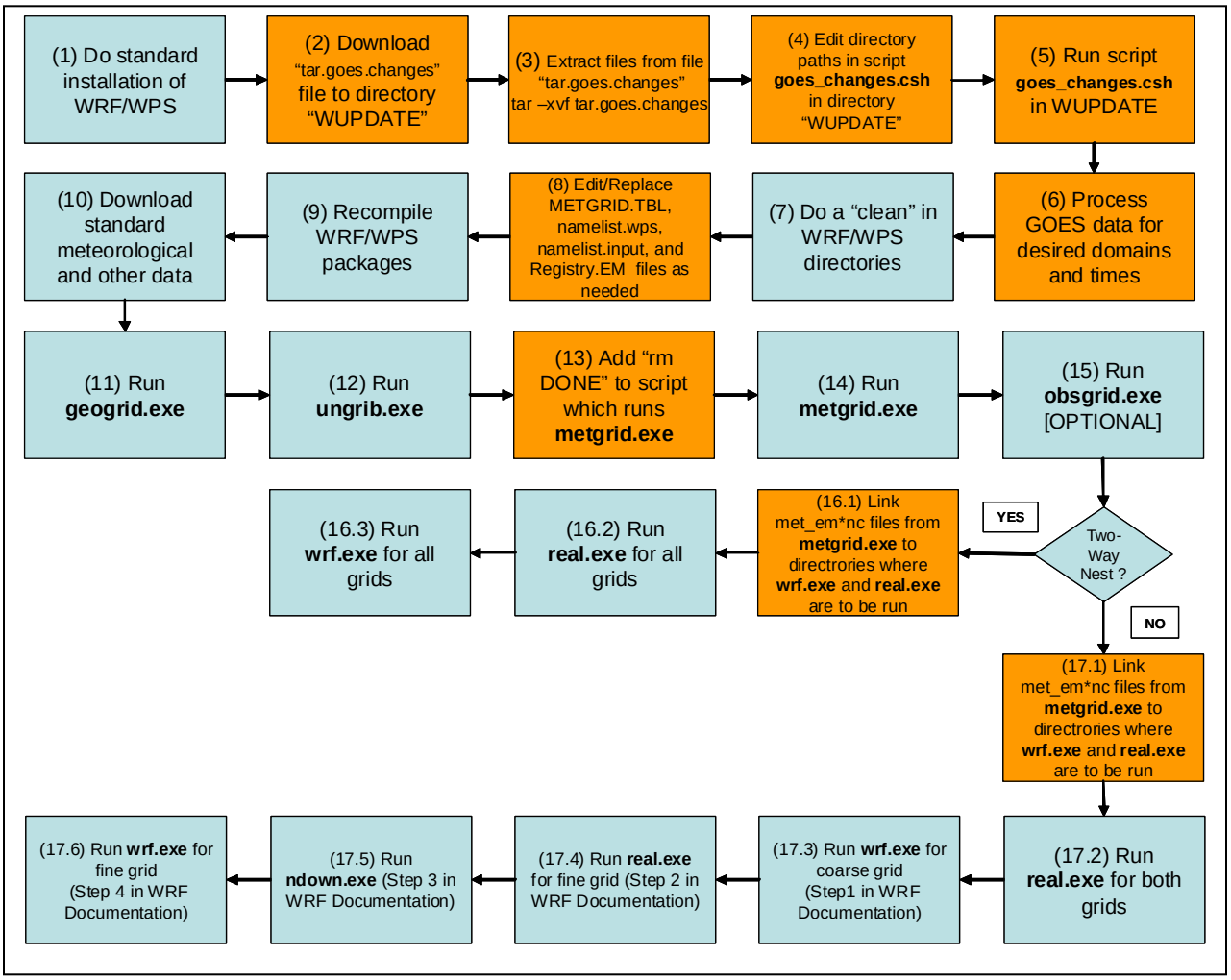
Step 6. Processing (or as the case may be, reprocessing) the GOES data with the regridding software. If the GOES data had previously been created without the updating described in 3.3, then they will have to be reprocessed.

Step 8 Possible editing of the **namelist.wps**, **namelist.input**, **METGRID.TBL**, and **Registry.EM** files. Example files **namelist.wps.GOES** and **namelist.input.GOES** (from the file **tar.goes.changes**) provide the context for editing the user's choices for those files with the information given in section 3. The script **goes_changes.csh** does a default updating of the **METGRID.TBL** and **Registry.EM** files based on the **METGRID.TBL.GOES** and **Registry.EM.GOES** files from the file **tar.goes.changes**. If that substitution is not adequate then the user would need to use the discussion in section 3 to make the necessary changes.

Step 13. Add the Unix command line "rm DONE" to the script which runs the program "metgrid.exe". The dummy file "DONE" is used to prevent file conflicts and allow processor number zero to finish processing the GOES data when multiple processors are running program **metgrid.exe**.

Steps 16.1, 17.1 Link the met_em*nc files to the locations where **real.exe** and **wrf.exe** are to be run.

Figure 5.1 Flowchart of the main steps in the WPS/WRF package with additional steps required to implement the processing of the GOES data. Blue boxes denote standard steps which are needed regardless of whether GOES data are being used or not. Orange boxes denote steps required to implement the processing of the GOES data. Step numbers are in parentheses. Program names are in bold case such as **metgrid.exe**.



6 Application Example

The results here are from a 30-km 72-h WRF run using the Kain-Fritsch cumulus parameterization from 00 UTC 4 February through 00 UTC 7 February of 2008 which includes the Super Tuesday tornado outbreak. This short case was chosen as an environment for creating and testing the GOES data implementation prior to using it on the June – September 2006 simulation.

6.1 Statistical Tools

Criteria are needed as what constitutes a "cloud" in the model framework. This is required because a number of model columns contain very small amounts of cloud water that would not be visible clouds in the real atmosphere. Precipitable cloud water (PCW) in units of mm is given by (6.1),

$$(6.1) \quad PCW = \frac{1000}{\rho_W} \int_{z_B}^{z_T} \left(10^{-3} \rho r_C\right) dz$$

where ρ_w is the density of liquid water (1000 kg m^{-3}), ρ is the density of air in units of kg m^{-3}, r_C is the cloud water mixing ratio in units of g kg^{-1}, and Z_G and Z_T are the terrain height and the top height of a given model column above mean sea level, respectively, in units of m. Sassen and Cho (1992) give a range of cloud optical thickness of $0.30 < \tau < 3.0$ for optical cirrus. Han et al. (2002) give the following formula relating PCW to effective droplet radius (Re), optical cloud thickness (τ), and the density of liquid water as shown in (6.2),

$$(6.2) \quad PCW = \frac{2}{3} R_E \tau \rho_W$$

which has units of kg m^{-2} but is numerically the same as mm of water. Using an effective droplet radius of 1 μ and a value of the cloud optical thickness in the middle of the opaque cirrus range of 1.0, gives a critical PCW value of 6.77 x 10^{-3} mm. Columns having PCW values greater than 6.77 x 10^{-3} mm where considered to be "clouds" for the comparison with the satellite data. From the satellite perspective clouds are defined by having cloud cover percentage based on the Imager cloud top pressure field greater than or equal to the values set for the namelist variable "cloud_threshold" set in file **namelist.input** as discussed in section 3.2. Based on these definitions of model and satellite clouds, a contingency table can be defined as shown in Table 6.1, where "H" is the number of grid cells where both model and satellite indicated cloud, "Z" is the number of grid cells where both model and satellite indicated clear conditions, "F" is the number of grid cells where the model indicated clouds and the satellite indicated clear conditions, and "M" is the number of grid cells where the model had clear conditions and the satellite indicated clouds. Based on these four cell counts, several statistics can be

Table 6.1 Contingency table categories for model-satellite cloud comparison. Same color scheme is used in the plots (a) and (b) in **Figure 6.3**

	Satellite Cloud	**Satellite No-Cloud**
Model Cloud	**H**	**F**
Model No-Cloud	**M**	**Z**

defined as given by equations 6.3-6.6 The false fraction in 6.3 gives the number of grid cells where the model indicated clouds and the satellite indicated clear conditions as a fraction of the total number of satellite observed clouds. The traditional bias statistic is given by 6.4. A disagreement index is defined by 6.5, and an analogous agreement index is defined by 6.6.

$$FF = \frac{F}{H + M} \tag{6.3}$$

$$BIAS = \frac{F + H}{H + M} \tag{6.4}$$

$$DI = \frac{M + F}{H + M + F + Z} \tag{6.5}$$

$$AI = \frac{H + Z}{H + M + F + Z} \tag{6.6}$$

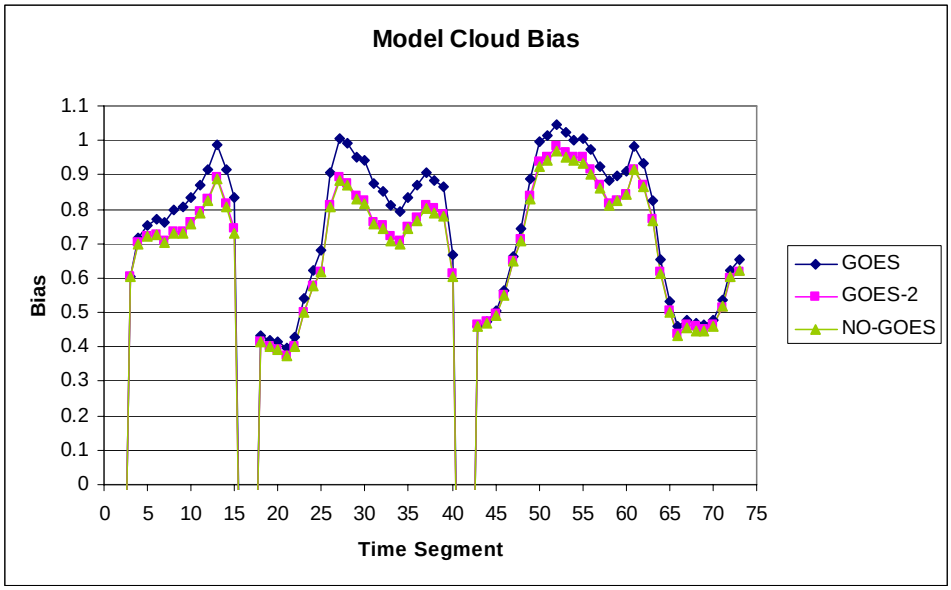
As an aside, many other verification statistics can be calculated from the cell counts from Table 6.1 (e.g. see McBride and Ebert 2000; Schaefer 1990). These include the probability of detection, the false alarm ratio, the accuracy for "non-events", the Hanssen-Kuipers score, and a modified critical success index which has become known as the equitable threat score. These have been used primarily in rainfall verification but they can be used for other variables as well.

6.2 Analysis of Test Run

Figure 6.1 and Figure 6.2 show the model cloud bias, false fraction, and agreement indices for three 72-h 30-km WRF runs. The three runs were the following: 1) a run labeled as "GOES" where the GOES data was used to suppress deep and shallow convection where the satellite indicated clear conditions, 2) a run labeled as "GOES-2" where the GOES data was used to suppress only deep convection where the satellite indicated clear conditions, and 3) a control run labeled as "NO-GOES" where the GOES data were not used. At first glance the "GOES" run seems to be an improvement over the control run as its bias values were generally closer to a value of one. However, examination of the false fraction numbers show higher values for the "GOES" run as compared to the "GOES-2" and "NO-GOES" runs, which were almost identical. The agreement index shows the same conclusion, with the "GOES-2" and "NO-GOES" runs having almost identical values and better performance than the "GOES" run. This leads to the conclusion that shallow convection performance was the main driver of the differences between the runs. Figure 6.3 shows horizontal plots for 0200 UTC 5 February 2008 (corresponds to time segment number 27 in Figure 6.1 and Figure 6.2). Plots (a) and (b) which use the color scheme from Table 6.1 show that the "GOES" run has more green areas (cell count "F", which is the model has clouds while the satellite

does not) than the "NO-GOES" run. These "green areas" were almost exclusively over portions of the Atlantic Ocean off the East coast, the Gulf of Mexico, and the Caribbean. Plots (c) and (d) show that these same maritime areas are mainly low clouds with large cloud top pressures and low cloud cover percentages. Deng et al. (2003), in describing the impact of the addition of the shallow convection algorithm to the Kain-Fritsch scheme, note that in marine cases the shallow convection plays an important role in drying the planetary boundary layer (PBL) beneath the shallow cumulus field through the subsidence around the cumulus updrafts. When that dynamic is removed in the "GOES" run, the model presumably is producing more grid-scale clouds at the top of the PBL. This conclusion, if true, also leads to two other issues for consideration in future research and analysis. The "GOES-2" and "NO-GOES" runs still show considerable "green areas" in the maritime areas described above. This behavior may be due to the fact that in the current Kain-Fritsch WRF code, the turbulent kinetic energy (TKE) is "hard-wired" as a constant value of 5.0 m^2s^{-2}. This was presumably done not to restrict the use of the Kain-Fritsch scheme in those cases where the user's PBL physics choice did not produce a TKE field. The second issue is that in addition to the "cloud" and "no-cloud" categories used for the model and satellite in constructing the contingency table in Table 6.1, a restriction could be added to remove the consideration of low clouds both in the model and the satellite. This is probably a good idea not only because of the shallow convection issues just discussed, but also because at times the GOES cloud products can have difficulty distinguishing between low-clouds and the surface.

While the GOES data were not benefical in this run, it is expected that they will have a significant impact on the summer June – September 2006 simulation where the synoptic forcing is much weaker.



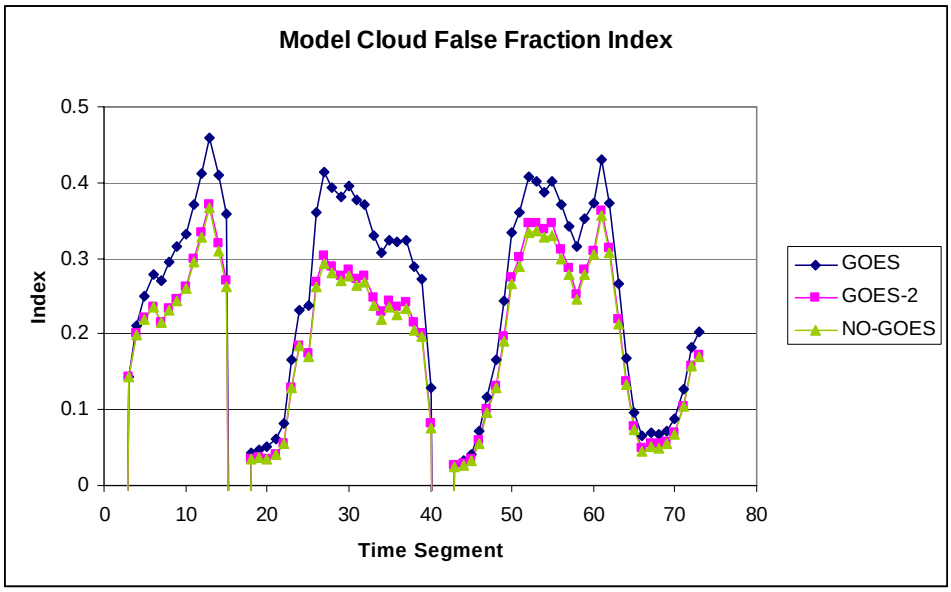


Figure 6.1 Model bias (top) and false fraction index (bottom) for the 72-h 30-km WRF simulation for the period 00 UTC 4 February through 00 UTC 7 February 2008. Run with convective suppression of deep and shallow convection in clear areas using GOES data is denoted by the blue curve, run with convective suppression of only deep convection in clear areas using GOES data is denoted by the purple curve, and with no GOES data is the green curve. Segment number 27 is the time 0200 UTC 5 Feb 2008 which is the time of the horizontal plots in Figure 6.3.

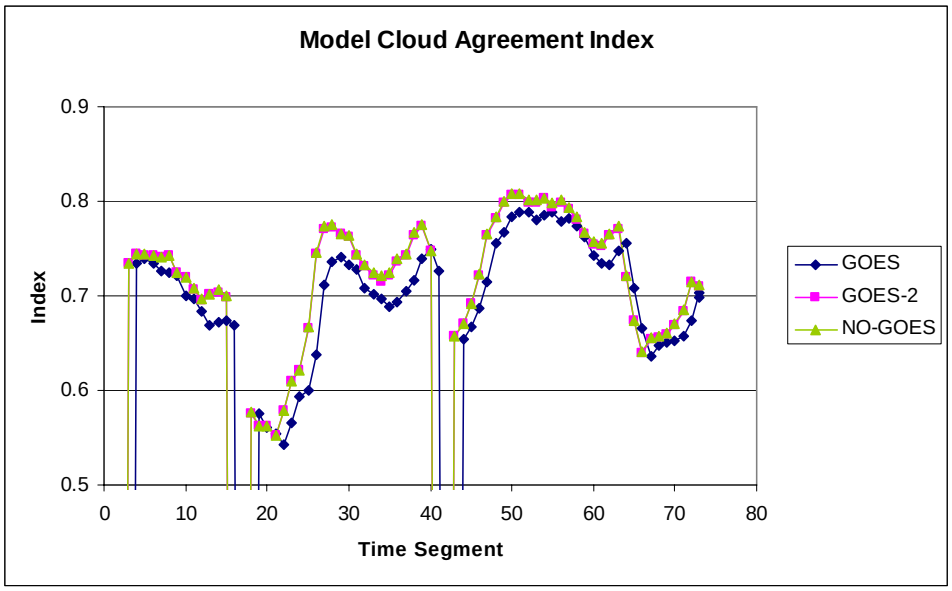


Figure 6.2 Model cloud agreement index for the 72-h 30-km WRF simulation for the period 00 UTC 4 February through 00 UTC 7 February 2008. Run with convective suppression of deep and shallow convection in clear areas using GOES data is denoted by the blue curve, run with convective suppression of only deep convection in clear areas using GOES data is denoted by the purple curve, and with no GOES data is the green curve. Segment number 27 is the time 0200 UTC 5 Feb 2008 which is the time of the horizontal plots in Figure 6.3.

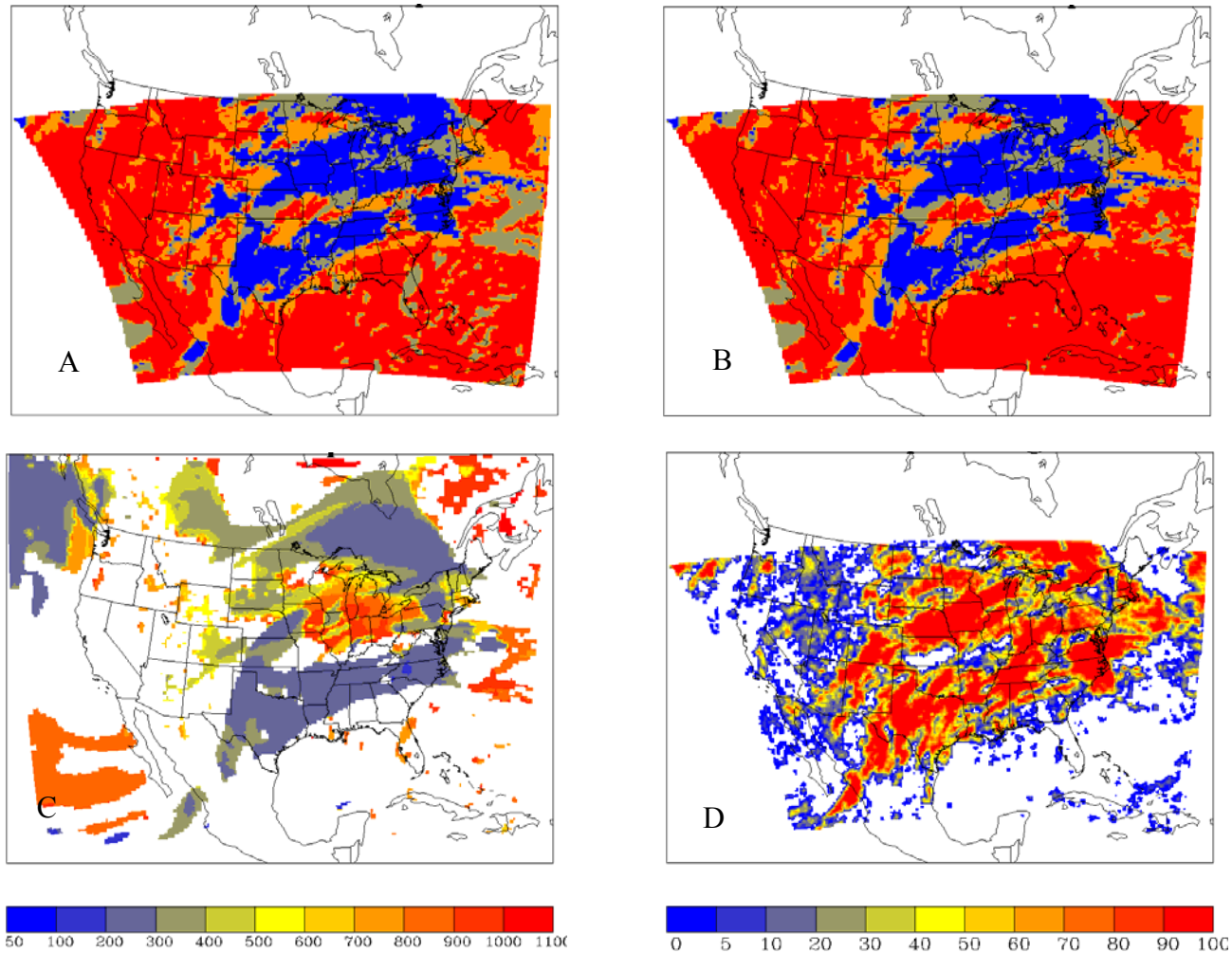


Figure 6.3 Plots for 0200 UTC 5 Feb 2008. Color code for (a) and (b): red, model and satellite clear, blue, model and satellite cloudy; green, model indicated cloud while satellite was clear; and orange, model indicated clear while satellite was cloudy (see **Table 6.1**). (a) Model-satellite comparison for case of using GOES data to control convection, (b) Model-satellite comparison for control case of not using GOES data to control convection, (c) model cloud top pressure (mb) for case of using GOES data to control convection, and (d) GOES cloud cover percentage with colored values greater than 1 %. See text for additional details.

7 References

Deng, A., N. L. Seaman, and J. S. Kain, 2003: A shallow-convection parameterization for mesoscale models. Part I: Submodel description and preliminary applications. *J. Atmos. Sci.*, **60**, 34-56.

Han, Q., W. B. Rossow, J. Zeng and R. Welch, 2002: Three Different Behaviors of Liquid Water Path of Water Clouds in Aerosol–Cloud Interactions. *J. Atmos. Sci.*, **59**, 726–735.

McBride, J. L. and E. E. Ebert, 2000: Verification of quantitative precipitation forecasts from operational numerical weather prediction models over Australia. *Wea. Forecasting*, **15**, 103-121.

Sassen, K. and B. S. Cho, 1992: Subvisual-Thin Cirrus Lidar Dataset for Satellite Verification and Climatological Research. *J. Appl. Meteor.*, **31**, 1275–1285.

Schaefer, J. T., 1990: The critical success index as an indicator of warning skill. *Wea. Forecasting*, **5**, 570-575.

APPENDIX B

Description of Variation Technique Used for Meeting the Target Vertical Velocity

1 Introduction

Section 2 gives a brief summary of the divergence adjustment software. Section 3 describes the control options specified in the namelist file "wdiag_options.inp". Section 4 describes the subroutine arguments to the main driver subroutine. Section 5 give a brief introduction to the basic variational equations. Section 6 provides some detail on applicable parts of O'Brien (1970). Section 7 applies the variational approach by O'Brien (1970) to the sigma-h coordinate system. Sections 8 and 9 give some of the details of the divergence adjustment process. Section 10 provides a series of flow charts depicting the "calling tree" of all the subroutines and functions in the software.

2 Brief Summary of WDIAG Module

The original software design for adjusting the three-dimensional divergence field (hereafter referred to as **WDIAG**) was done in conjunction with the RAMS model, which uses the sigma-h vertical coordinate (described in section 7). Therefore, the core calculations where all performed in the sigma-h system. This same approach was maintained with later versions which ingested MM5 data which uses the sigma-p vertical coordinate. As a final step the results in the sigma-h system were then interpolated back to the sigma-p system. An early decision was made for this current version which ingests WRF data (which also uses the sigma-p vertical coordinate) to keep the same approach. This was done primarily for two reasons. The first was to avoid the necessity of code changes for a different vertical coordinate. The second reason is that in the sigma-h system, there is a simple relationship between the physical vertical velocity (w=dz/dt) and the sigma-h "vertical velocity" (dh/dt). In the sigma-p system, the relationship between the physical vertical velocity (w=dz/dt) and the sigma-p "vertical velocity" requires an instantaneous pressure tendency. This variable is not part of the default WRF output history and is generally not well resolved with the hourly output history files we typically utilize.

Table 2.1 shows the names of the twelve subroutines, in the order called, within the main driver subroutine of **WDIAG**. Names associated with program units and files will be in **bold** print. Each will be mentioned and briefly discussed. More details on the code structure are given in section 10. For the first time **WDIAG** is called (when the time index "nt" is equal to 1) **input_wdiag_options** is called to read the options which control the processing found in the namelist file "**wdiag_options.inp**". The routine **t_from_theta** is then called which calculates the three-dimensional field of temperature from the WRF input field of perturbation potential temperature. Next, the routine **est_mavail** is called which estimates the surface moisture availability, which is used later for the vertical interpolation process within the surface layer. The next three routines which are called are for the first entry into **WDIAG** (time index "nt" is equal to 1). These routines are **choose_grids**, **allocate_arrays_wdiag_step2**, and **calc_filter_weights**. The routine **choose_grids** sets up the latitude-longitude grids and their corresponding cartesian grids using the WRF Lambert-Conformal map projection.

The routine **allocate_arrays_wdiag_step2** allocates some of the arrays for the sigma-p system and arrays which do not change their dimensions. The routine **calc_filter_weights** calculates the two-dimensional filtering weights if that option is selected in the file "**wdiag_options.inp**". The next routine to be called, **interp_horz_data**, performs one of two functions depending on a choice made in the file "**wdiag_options.inp**". If the namelist variable **use_fine_grid** (discussed in section 3.1) is set to ".false.", then the main function of **interp_horz_data** is simply to transfer the input data from the "in arrays" to the "local arrays". On the other hand, if the namelist variable **use_fine_grid** is set to ".true.", then the main function of **interp_horz_data** is to perform horizontal interpolation to a fine-mesh grid which has a smaller mesh size than the original WRF grid. The routine **get_height_grids** is then called which calculates all the height grids in the WRF sigma-p system. Next, the routine **get_sigmah_grid** is called which calculates the sigma-h height grid which is used for all the core **WDIAG** calculations. The final interpolation step for the incoming data is then done by the routine **model_to_sigmaz,** which performs vertical interpolation of the WRF data from the sigma-p system to the sigma-h system. All the divergence calculations are then done by calling **do_wind_adj**. The final step in the **WDIAG** driver is to deallocate the sigma-h arrays by calling **deallocate_interp_arrays**.

Table 2.1 Main subroutine calls from subroutine **WDIAG**

WDIAG Subroutine Name	Description
input_wdiag_options	for nt=1, read namelist file "wdiag_options.inp"
t_from_theta	calculate temperature from perturbation theta
est_mavail	estimate surface moisture availability from surface fluxes
choose_grids	for nt=1, determine latitude-longitude and other grids
allocate_arrays_wdiag_step2	for nt=1, allocate arrays for sigma-p system
calc_filter_weights	for nt=1, calculate two-dimensional filter weights
interp_horz_data	horizontally interpolate data to a different grid
get_height_grids	calculate height grids in the sigma-p system
get_sigmah_grid	calculate height grids in the sigma-h system
model_to_sigmaz	vertically interpolate data to the sigma-h grid
do_wind_adj	perform divergence adjustment
deallocate_interp_arrays	deal locate sigma-h arrays

3 Namelist File "wdiag_options.inp" Options

There are five sections in the file "wdiag_options.inp" which control options for the **WDIAG** software. These are summarized in Table 3.1 through Table. 3.5. Each variable of each section will now be discussed in order.

3.1 Grid, interpolation, and analysis options

Options for this section are given in Table 3.1. The variable **use_fine_grid** determines whether the **WDIAG** calculations will be performed on the original WRF Arakawa C-grid (**use_fine_grid** = ".false."), or on a finer mesh (**use_fine_grid** = ."true."). The ramifications of this choice and a diagram of the Arakawa C-grid are given in section 9.1. If **use_fine_grid** = ".true.", then the horizontal resolution of the fine mesh grid in km is set by the variable **dxg_fine**. A typical setting for this would be half of the original WRF resolution. For example, if the original WRF horizontal mesh size was 36 km, then an appropriate choice for **dxg_fine** would be 18 km or less. If **use_fine_grid** = ".true.", then there are two horizontal interpolation choices given by the variable **hz_interp_type**. If **hz_interp_type** is equal to "BILINEAR", then simple bilinear interpolation is used. . If **hz_interp_type** is equal to "BICUBIC", then bicubic splines are used to interpolate to the fine mesh grid. The variable **hz_interp_type_out** controls the type of horizontal interpolation used for the variables which are output from **WDIAG**. If **use_fine_grid** = ".false.", then the two choices are "BILINEAR" or BICUBIC", and have the same meaning as given for the variable **hz_interp_type**. This option is needed in this case to interpolate certain wind components which in the **WDIAG** process are defined at the scalar grid points and are needed at the u or v WRF stagger points. If **use_fine_grid** = ".true.", then the three choices for **hz_interp_type_out** are "BILINEAR", "BICUBIC", or "GAUSSIAN". Again the first two choices have the same meaning as before. When **interp_type_out** is "GAUSSIAN", an area averaging is done using exponential weights to obtain a value on the output grid from the fine grid. The parameters which control this option will be discussed in Table 3.2. The five variables **dz_sig_bot**, **dz_sig_max**, **dz_sig_rat**, **sigma_h_stop**, and **kabove_set** define the sigma-h grid. The variable **dz_sig_bot** sets the thickness in meters of the lowest sigma-h layer. It should less than or equal to half the minimum sigma-p thickness. If this constraint is violated the program stops with a corresponding message. The variable **dz_sig_max** sets the maximum thickness in meters of the largest sigma-h layer. It should be smaller than the largest sigma-p layer. The variable **dz_sig_rat** must be greater than one and is the ratio by which layers are increased in size from the surface upwards until the value of **dz_sig_max** is met**.** If the variable **sigma_h_stop** is set to ".true.", then the program is stopped in subroutine **get_sigmah_grid** and the user can examine the grid setup and make changes as desired. Since the top of the sigma-p domain is not flat, and the top of the sigma-h domain is, it is desirable to have a few layers in the sigma-h system be above the highest sigma-p height. This accomplished by the variable **kabove_set** which determines the number of sigma-h layers of thickness **dz_sig_max** above the highest sigma-p height. In the sigma-h system, one or more levels may exist below the lowest sigma-p level. If the variable **no_sfc_influence** is set to ".false.", then similarity theory is used to vertically interpolate temperature, water vapor, and the u and v wind components. If the variable **no_sfc_influence** is set to ".true.", then all sigma-h levels below the lowest sigma-p level use the values from the latter level. The four variables **smooth_uv**, **smooth_den**, **nxy_filter**, and **wave_min_fac_dxdy** control the Lanczos two-dimensional filtering options (Duchon 1979) for the density and horizontal winds. Filtering is probably needed only when the fine grid option is chosen. Its purpose is to remove $2\Delta x$ noise in the divergence fields. If **smooth_uv** is set to ".true.", then filtering is done on

the horizontal wind components. . If **smooth_density** is set to ".true.", then filtering is done on the density field. The variable **nxy_filter** is the "one-sided" size of the filtering box that is used at each grid location. The total number of grid points in the box is accordingly (2 nxy_filter+1)*(2 nxy_filter+1). The variable **wave_min_fac_dxdy** controls the wavelength of the features to be removed. Its influence depends on the actual grid being used. For example, if **use_fine_grid** = ".false." and the horizontal mesh size is 36 km, a value of **wave_min_fac_dxdy** equal to 2.5 will remove all features having wavelengths less than or equal to 2.5 times the mesh size or 90 km. On the other hand, if **use_fine_grid** = ".true." and the horizontal mesh size is 18 km, a value of **wave_min_fac_dxdy** equal to 5.0 would be needed to remove features of 90 km. The variable **use_exp_weights** is used in the vertical interpolation in going from the sigma-h grid back to the sigma-p grid. If the variable **use_exp_weights** = ".true.", then the sigma-h layers closest to the sigma-p level will be weighted exponentially more heavily when the sigma-h grid has a finer vertical resolution than the sigma-p grid. If the variable **use_exp_weights** = ".false.", then the sigma-h layers are weighted proportionally to their thicknesses. If the variable **use_exp_weights** = ".true.", then the variable **wmin_exp** is the minimum weight for the sigma-h layers in question. The variable **use_density_tendency** determines how the density tendency term is treated in the continuity equation in the sigma-h coordinate system. Neither choice here is totally acceptable. If **use_density_tendency** is ".false.", then the density tendency is set to zero. . If **use_density_tendency** is ".true.", then it is diagnosed from known terms. The density tendency is calculated before the divergences are adjusted to meet the specified targets. It is then included with the adjusted divergences in calculating the final vertical motion field. The final variable in this section, **output_on_uv_stagger**, controls the grid for the output winds. If **output_on_uv_stagger** is ".true.", then the winds will be interpolated to the corresponding WRF Arakawa C-grid u and v grid locations. If **output_on_uv_stagger** is ".false.", then the winds will be interpolated to the corresponding WRF Arakawa C-grid scalar grid locations.

Table 3.1 Grid, interpolation, and analysis options in the "**wdiag_options.inp**" file.

use_fine_grid=	.false.	If .true., use finer grid mesh than original
dxg_fine =	18.0	typically half original mesh size in km
hz_interp_type =	'BILINEAR'	BILINEAR or BICUBIC
hz_interp_type_out =	'BILINEAR'	BILINEAR, BICUBIC, or GAUSSIAN
dz_sig_bot =	3.0	thickness of bottom sigma-h layer in m
dz_sig_max =	300.0	maximum layer thickness in m for interpolation sigma-h grid
dz_sig_rat =	1.2	layer ratio for interpolation sigma-h grid
sigma_h_stop =	.false.	if true, stop after making sigma-h grid
kabove_set =	3	number of levels added to WRF grid in sigma-h interpolation grid
no_sfc_influence =	.false.	If true, all sigmah levels below level=1 sigmap take on those values
smooth_uv =	.false.	smooth horizontal winds
smooth_den =	.false.	smooth density
nxy_filter =	8	number 1-sided filtering points
wave_min_fac_dxdy =	2.5	cutoff wavelength in grid units
use_exp_weights =	.false.	if true, weight layers with exponential weights
wmin_exp =	0.50	minimum weight when use_exp_weights is true
use_density_tendency =	.true.	if true, diagnose current model density tendency from known terms
output_on_uv_stagger =	.false.	if true, output winds on u-v stagger, if false, output on scalar grid

3.2 Parameters used when hz_interp_type_out = 'GAUSSIAN'

Options for this section are given in Table 3.2. These options apply when **use_fine_grid** = ".true." and when **_interp_type_out** is "GAUSSIAN". The variables **fname_wgts_fine_to_crs_model**, **fname_wgts_fine_to_u_model**, and **fname_wgts_fine_to_v_model** are the file names for the respective weights to go from the fine grid to one of the WRF staggered grids. The variables **wgtg_min** and **dist_fac_wgtg_min** control the calculation of the exponential weights. The weights will range from a weight of one at a distance of zero to a minimum weight of **wgtg_min** at the distance of **dist_fac_wgtg_min** times the original WRF horizontal resolution. So, for example, if **dist_fac_wgtg_min** equals 1.5, **wgtg_min** equals 10^{-3}, and the original WRF resolution is 36 km, the minimum weight will be 10^{-3} at a distance of 54 km.

Table 3.2 Parameters used when **hz_interp_type_out** = 'GAUSSIAN' in the "**wdiag_options.inp**" file.

fname_wgts_fine_to_crs_model=	'fine_wgts_crs.dat'	weight file to go from fine to crs (GAUSSIAN)
fname_wgts_fine_to_u_model =	'fine_wgts_u.dat'	weight file to go from fine to u (GAUSSIAN)
fname_wgts_fine_to_v_model =	'fine_wgts_v.dat'	weight file to go from fine to v (GAUSSIAN)
wgtg_min =	1.0e-03	for hz_interp_type_out = GAUSSIAN, minimum
dist_fac_wgtg_min =	1.5	for hz_interp_type_out = GAUSSIAN, distance factor for wgtg_min

3.3 Parameters for analytical w target

Options for this section are given in Table 3.3. In the debugging phase of **WDIAG,** it was useful to supply a simple "bulls-eye" vertical velocity target to check for coding errors. This option has been kept in case further changes are made to the code and basic checks want to be accomplished. If the variable **use_analytical_target** is set to ".true.", then the simple analytical target will be used rather than the one supplied through the input arguments to **WDIAG** (described in section 4). The center of the vertical motion target is set as a fraction of the west-east, north-south, and vertical domain sizes of the three-dimensional domain by **xa_frac**, **ya_frac**, and **za_frac**, respectively. For example, if the latter three variables are all set to 0.50 then the target will be approximately centered horizontally and vertically within the WRF domain. The size of the target pattern is controlled exponentially by two variables in each direction. For example, for the west-east direction, the target is reduced in magnitude by the weight **xscale_wgt** at the distance of **xscale_frac** times the west-east domain size. Control in the other directions is accomplished in a similar manner by the variables **yscale_frac**, **yscale_wgt**, **zscale_frac**, and **zscale_wgt**. The maximum vertical motion target in m s^{-1} is given by **wmax_anl_target**.

Table 3.3. Parameters for analytical w target in the "**wdiag_options.inp**" file.

use_analytical_target =	.false.	use analytical target w
xa_frac =	0.50	x location of maximum of analytical target as a fraction of W-E domain
ya_frac =	0.50	y location of maximum of analytical target as a fraction of N_S domain
za_frac =	0.05	z location of maximum of analytical target as a fraction of vert. depth
xscale_frac =	0.10	x-scale as fraction of W-E domain
xscale_wgt =	0.10	fraction of analytical maximum at distance xscale_frac
yscale_frac =	0.10	y-scale as fraction of N-S domain
yscale_wgt =	0.10	fraction of analytical maximum at distance yscale_frac
zscale_frac =	0.10	z-scale as fraction of vertical depth
zscale_wgt =	0.10	fraction of analytical maximum at distance zscale_frac
wmax_anl_target =	0.50	analytical maximum vertical velocity in m/sec

3.4 Parameters for Sequential Over-Relaxation

Options for this section are given in Table 3.4. The reader is referred to Press et al. 1989 for theoretical details on the sequential over-relaxation (SOR) methodology. This section provides control over the parameters that influence the SOR calculations. The variable **omega_div** is essentially a weighting factor to determine how much the new estimate is to be weighted. For convergence it must be in the range of 0.0 < **omega_div** < 2.0. The speed of convergence of the SOR procedure is very sensitive to this parameter. Theoretically the optimal value is a function of the unknown field being determined. But the size of the grid and the horizontal grid mesh size are also key players in the optimal value. The variables **calc_omega**, **omega_min**, **omega_max**. and **niter_omega** enable one to experiment with a range of values for **omega_div** and examine the results before doing production runs. This option is activated by setting **calc_omega** to ".true.", setting the minimum and maximum values to be considered by **omega_min** and **omega_max**, and the number of intervals between the latter two values by **niter_omega**. When this feature is activated the program stops when all the combinations have been tested. The boundary condition type for the SOR calculations is set by the variable **div_sor_bc_type**. When it is set to "D", Dirichlet boundary conditions set values to zero on the exterior boundaries of the domain. When it is set to "N", the Neumann boundary conditions set the exterior values in a way that is consistent with the wind component perpendicular to the edge in question. The Dirichlet boundary conditions have typically been used since they minimize the kinetic energy of the divergent component of the wind (Lynch 1989). The maximum and minimum number of iterations for the SOR process is set by the variables **maxit** and **minit**, respectively. The error threshold of convergence is determined by the variables **u_div_scale** and **u_div_per**. Details on how these parameters are used to calculate the convergence threshold are given in section 8.

Table 3.4. Parameters for Sequential Over-Relaxation (SOR) calculations in the "**wdiag_options.inp**" file.

omega_div =	1.965	omega value for div SOR, must be in the range 0.0 < omega_div < 2.0
calc_omega =	.false.	iterate to find optimal omega_div value
omega_min =	1.50	minimum value for omega_div when calc_omega = .true.
omega_max =	1.99	maximum value for omega_div when calc_omega = .true.
niter_omega =	20	number of iterations when calc_omega = .true.
div_sor_bc_type =	'D'	boundary condition type: D for 'dirichlet' or N for 'neumann'
maxit =	3000	maximum number of iterations for SOR
minit =	10	minimum number of iterations for SOR
u_div_scale =	1.00	velocity scale in m/sec to help calculate potential field convergence
u_div_perr =	1.00	acceptable percentage error of u_div_scale

3.5 Parameters used in subroutine check_vertical_velocities

Options for this section are given in Table. 3.5. The user is referred to section 9 on how these are implemented.

Table. 3.5 Parameters used in subroutine **check_vertical_velocities** to control maximum change in column divergences in the "**wdiag_options.inp**" file.

wind_speed_min =	10.0	min wind speed parameter in m/sec for checking vertical velocities
wind_speed_max =	40.0	max wind speed parameter in m/sec for checking vertical velocities
div_per_max_lim =	75.0	limit for the maximum column value of div_per to stop iterations
div_per_iter_max =	100	maximum number of iterations to adjust large target vertical velocities
fz_min =	0.10	height weight factor at the top for checking vertical velocities
fz_max =	1.00	height weight factor at the sfc for checking vertical velocities

4 Arguments to WDIAG Driver Subroutine

Table 4.1 shows the argument list for the main driver of the **WDIAG** software. Most of the input entries should be self evident within the context of the WRF modeling system. Brief comments will be made in this section to clarify other issues.

The first argument, **nt**, is the time counter or index. The first time the code is called it must be equal to one. This tells the code, among other things, to allocate arrays which will not change in size. After this first time, the argument **nt** must be different from a value of one. Arguments 2 through 7, which are, respectively, **imax_in**, **jmax_in**, **kmax_in**, **imaxr_in**, **jmaxr_in**, and **kmaxr_in**, control the array dimensions. The variables **imaxr_in**, **jmaxr_in**, and **kmaxr_in** are the dimensions of the WRF Arakawa C-grid scalar, half-sigma grid. The variables **imax_in**, **jmax_in**, and **kmax_in** are the maximum dimensions of any of the Arakawa C-grids. So, relative to the scalar, half-sigma grid dimensions, **imax_in** = **imaxr_in**+1, **jmax_in**=**jmaxr_in**+1, and **kmax_in** = **kmaxr_in**+1. All the two-dimensional (2D) arrays are dimensioned by (**imax_in**, **jmax_in**), and all three-dimensional (3D) arrays are dimensioned by (**imax_in**, **jmax_in**, **kmax_in**). The **z_in** array (argument 8) is the height grid for the scalar, half-sigma grid.

It can be obtained from the geopotential variables in the WRF output history. However, the latter variables are with respect to the full-sigma grid. Equation (4-1) below illustrates the averaging process needed to obtain the **z_in** array. The geopotential variables are denoted by Φ, subscript "B" refers to base values, subscript "P" refers to perturbation values, superscripts of k,F and k+1,F are in the full-sigma grid system, and g is the acceleration of gravity.

(4-1) $$Z_{k,H} = \frac{0.50}{g}\left(\phi_B^{k,F} + \phi_B^{k+1,F} + \phi_P^{k,F} + \phi_P^{k+1,F}\right)$$

Arguments 9 and 10, **w_target_in** and **z_target_in**, are 2D arrays which help define the target vertical velocities and their height. Locations where no changes are need to be set to the value of -9999.0. The array **w_target_in** contains the target physical velocity (dz/dt) in units of m s^{-1}, and the array **z_target_in** contains the height of this target (meters above mean sea level). The vertical velocity target definition is completed by arguments 54 and 55, which are **zlev_wadj_bot_in** and **zlev_wadj_top_in**. These latter two variables are the bottom and top heights (meters above mean sea level), respectively, of the cloud layer either to be removed (negative values of **w_target_in**) or created (positive values of **w_target_in**).

Arguments 11 through 35 are input variables obtained from reading a WRF output history file. The only change from the WRF values is that pressure variables have been converted to units of mb, and the water vapor mixing ratio has been converted to units of g kg^{-1}.

Arguments 36 through 47 are output variables. Arguments 36 through 39 are **chi_out**, **div_out**, **udiv_old**, and **vdiv_old** and describe the existing divergence field before any adjustments are made. The argument **chi_out** is the potential field (discussed in section 8) in units of kg s^{-1} and reduced by a factor of 10^{-4}. The argument **div_out** is the original horizontal mass divergence in units of kg m^{-3} s^{-1}. The arguments **udiv_old** and **vdiv_old** are the original horizontal divergent wind components. These latter two variables can be output either on their respective u-v stagger or on the scalar grid, depending on the choice made by the user as described in section 3.1. These same variables after adjustment are arguments 40 through 43, are named **chi_new_out**, **div_corr_out**, **udiv_new_out**, and **vdiv_new_out**. The final new output wind field is given by arguments 44 through 46, and are named **unew_out**, **vnew_out**, and **wnew_out**. As previously mentioned with earlier output wind components, **unew_out** and **vnew_out** can be output either on their respective u-v stagger or on the scalar grid, depending on the choice made by the user. The remaining output variable is the array **diter_out**. It returns the number of iterations needed to satisfy equation (8-7).

Table 4.1. Argument list for main driver of **WDIAG**. The columns, from left to right, are: 1) argument number N; 2) argument name;3) dimension (0D, 1D, 2D, or 3D); 4) IO – representing IN/OUT; 5) description; 6) HGRID – horizontal grid type (relative to Arakawa C-grid, CRS for scalar-grid, U for u-grid, and V for v-grid; 7) VGRID – vertical grid type – either half or full, and 8) Units/Comments. In column 8, if the units are different from WRF units or expected units the scaling factor is in square brackets.

4.1.1.1.	4.1.1.1.2	4.1.1.1.3	4.1.1.1.4	**4.1.1.2 DESCRIPTION**	4.1.1.2.1	4.1.1.2.2	4.1.1.2.3 Units/Comments
1	nt	0D	IN	time index (first time in, nt = 1)			
2	imax_in	0D	IN	west-east dimension of WRF u grid	U		
3	jmax_in	0D	IN	north-south dimension of WRF v grid	V		
4	kmax_in	0D	IN	number of vertical levels for full-sigma grid		full	
5	imaxr_in	0D	IN	west-east dimension of WRF scalar grid	CRS		
6	jmaxr_in	0D	IN	north-south dimension of WRF scalar grid	CRS		
7	kmaxr_in	0D	IN	number of vertical levels for half-sigma grid		half	
8	z_in	3D	IN	height grid	CRS	half	meters
9	w_target_in	2D	IN	w target	CRS		meters/sec
10	z_target_in	2D	IN	z target	CRS		meters
11	thetap_in	3D	IN	perturbation potential temperature	CRS	half	K
12	pp_in	3D	IN	perturbation pressure	CRS	half	mb [1.0e-02]
13	pb_in	3D	IN	base_pressure	CRS	half	mb [1.0e-02]
14	q_in	3D	IN	water vapor mixing ratio	CRS	half	g/kg [1000]
15	u_in	3D	IN	u wind	U	half	meters/sec
16	v_in	3D	IN	v wind	V	half	meters/sec
17	w_in	3D	IN	w wind	CRS	full	meters/sec
18	glat_in	2D	IN	latitude	CRS		degrees
19	glon_in	2D	IN	longitude	CRS		degrees
20	ter_in	2D	IN	terrain	CRS		meters
21	mapscale_in	2D	IN	mapscale	CRS		
22	mapscale_u_in	2D	IN	mapscale	U		
23	mapscale_v_in	2D	IN	mapscale	V		

24	mu_in	2D	IN	perturbation dry mass	CRS		mb [1.0e-02]
25	mub_in	2D	IN	base dry mass	CRS		mb [1.0e-02]
26	tsfc_in	2D	IN	sfc temperature	CRS		K
27	xland_in	2D	IN	land/water mask	CRS		1-land,2-water
28	ustar_in	2D	IN	ustar	CRS		meters/sec
29	zpbl_in	2D	IN	height pbl	CRS		meters
30	hflux_in	2D	IN	heat flux	CRS		Watts/m**2
31	mflux_in	2D	IN	latent fluxe	CRS		Watts/m**2
32	zr_in	2D	IN	roughness	CRS		meters
33	psfc_in	2D	IN	surface pressure	CRS		mb [1.0e-02]
34	znu_in	1D	IN	half-sigma grid	none		
35	znw_in	1D	IN	full-sigma grid	none		
36	chi_out	3D	OUT	original potential. field	CRS	half	kg/sec [1.0e-04]
37	div_out	3D	OUT	original divergence	CRS	half	kg/m**3/sec
38	udiv_old	3D	OUT	original udiv	CRS/U	half	meters/sec
39	vdiv_old	3D	OUT	original vdiv	CRS/V	half	meters/sec
40	chi_new_out	3D	OUT	new potential field	CRS	half	kg/sec [1.0e-04]
41	div_corr_out	3D	OUT	new divergence	CRS	half	kg/m**3/sec
42	udiv_new_out	3D	OUT	new udiv	CRS/U	half	meters/sec
43	vdiv_new_out	3D	OUT	new vdiv	CRS/V	half	meters/sec
44	unew_out	3D	OUT	new u	CRS/U	half	meters/sec
45	vnew_out	3D	OUT	new v	CRS/V	half	meters/sec
46	wnew_out	3D	OUT	new w	CRS	full	meters/sec
47	diter_out	2D	OUT	number of Div. Passes			
48	dxg_in	0D	IN	horizontal grid mesh in km			km [1.0e-03]
49	wrf_lc_lat_n_in	0D	IN	Lambert-Conformal North Latitude			degrees
50	wrf_lc_lat_s_in	0D	IN	Lambert-Conformal d South Latitude			degrees
51	wrf_glatc_in	0D	IN	pole Latitude for all grids (x=0,y=0)			degrees
52	wrf_glon_in	0D	IN	pole Longitude for all grids (x=0,y=0)			degrees
53	ptop_wrf	0D	IN	pressure at top sigma-p level			mb [1.0e-02]
54	zlev_wadj_bot_in	2D	IN	bottom of adjustment layer	CRS		meters
55	zlev_wadj_bot_in	2D	IN	top of adjustment layer	CRS		meters
56	lulog_in	0D	IN	logical unit for log file			

5 Basic Variational Equations

This section gives a brief introduction to the basic variational equations. Section 6 takes these principles and supplements the development of certain equations in O'Brien (1970). Section 7 applies the same approach to the sigma-h coordinate system. Section 8 provides further details on the divergence process. Section 9 gives a brief description of the basic divergence calculation process.

This discussion follows after Chapter 17 of Arfken (1985) on the calculus of variations. The calculus of variations, in its simplest form, involves the minimization of an integral as given by (5-1):

(5-1) $$J = \int_{x_1}^{x_2} f\left(y, y_x, x\right) dx, \; where \; y = y(x), \; and \; y_x = \frac{dy}{dx}$$

where f is a known function of y, dy/dx, and x. In this context, y, dy/dx, and x all are treated as independent variables and the specific function y(x) is unknown. In other words, whereas the limits of integration (x_1 and x_2) are known, the integration path is not known. The ultimate goal is determining the integration path which minimizes J. To accomplish this, another function, η, is introduced with the only restrictions being that it is differentiable and satisfies the boundary conditions as given by (5-2).

(5-2) $$\eta(x_1) = \eta(x_2) = 0$$

The function y(x) is then constructed as in (5-3), as a function of x, η(x), and a scale factor α.

(5-3) $$y(x, \alpha) = y(x, 0) + \alpha\, \eta(x)$$

The original minimization integral in (5-1) then becomes (5-4), with the minimum value obtained by taking the partial derivative with respect to the scale factor α and setting the derivative equal to zero as in (5-5).

(5-4) $$J = \int_{x_1}^{x_2} f\left(y(x, \alpha), y_x(x, \alpha), x\right) dx$$

(5-5) $$\left[\frac{\partial J(\alpha)}{\partial \alpha} \right]_{\alpha = 0} = 0$$

Using the chain rule of differentiation the derivative given by (5-5) is expanded as given by (5-6).

(5-6) $$\frac{\partial J(\alpha)}{\partial \alpha} = \int_{x_1}^{x_2} \left[\frac{\partial f}{\partial y}\frac{\partial y}{\partial \alpha} + \frac{\partial f}{\partial y_x}\frac{\partial y_x}{\partial \alpha} \right] dx$$

Taking the necessary derivatives of (5-3) we obtain the derivatives as given by (5-7) and (5-8) which are needed in (5-6).

(5-7) $$\frac{\partial y(x,\alpha)}{\partial \alpha} = \eta(x)$$

(5-8) $$\frac{\partial y_x(x,\alpha)}{\partial \alpha} = \frac{d\eta(x)}{dx}$$

Substituting (5-7) and (5-8) into (5-6) gives the expanded version of the minimization derivative as in (5-9). These substitutions give rise to two terms in the integrand, labeled as "1" and "2". Term one can not be simplified, but term two can be integrated by parts as shown in (5-10). This integration also produces two terms, labeled as "3" and "4". Term three becomes zero because of the zero boundary conditions specified for η(x) in (5-2). The end result of these substitutions and calculations is that the original integrand is reduced to terms one and four as in (5-11).

(5-9) $$\frac{\partial J(\alpha)}{\partial \alpha} = \int_{x_1}^{x_2} \left[\underbrace{\frac{\partial f}{\partial y}\eta(x)}_{1} + \underbrace{\frac{\partial f}{\partial y_x}\frac{d\eta(x)}{dx}}_{2} \right] dx$$

(5-10) $$\int_{x_1}^{x_2} \frac{\partial f}{\partial y_x}\frac{d\eta(x)}{dx}\,dx = \underbrace{\eta(x)\frac{\partial f}{\partial y_x}\Bigg|_{x_1}^{x_2}}_{3} - \underbrace{\int_{x_1}^{x_2} \eta(x)\frac{d}{dx}\frac{\partial f}{\partial y_x}\,dx}_{4}$$

(5-11) $$\int_{x_1}^{x_2} \left[\frac{\partial f}{\partial y} - \frac{d}{dx}\frac{\partial f}{\partial y_x} \right] \eta(x)\,dx = 0$$

Apart from the boundary conditions in (5-2), η(x) is arbitrary and therefore the integral in (5-11) is zero only when the integrand is zero, as in (5-12), which is called the Euler equation.

(5-12) $$\frac{\partial f}{\partial y} - \frac{d}{dx}\frac{\partial f}{\partial y_x} = 0$$

Although the details will not be given here, the Euler equation in (5-12) can be expanded to include several dependent and independent variables as in (5-13), where we now have an Euler equation for each y_i and a sum over the independent variables x_j.

(5-13) $$\frac{\partial f}{\partial y_i} - \sum_j \frac{\partial}{\partial x_j}\frac{\partial f}{\partial y_{ij}} = 0, \; where \; y_{ij} = \frac{\partial y_i}{\partial x_j}$$

Sometimes the y_i values are not independent of each other which introduces the idea of one or more constraints (indexed with respect to k) as in (5-14).

(5-14) $$\varphi_k\left(y_i, x_j\right) = 0$$

The reader is referred to Arfken (1985) for details, but the end result of incorporating the constraint involves a new function "g" as in (5-15), which is the sum of our previous function "f" and one or more constraints φ_k each multiplied by a Lagrangian multiplier λ_k.

(5-15) $$g = f + \sum_k \lambda_k \varphi_k$$

The set of Euler equations in (5-13) with the addition of constraints then becomes (5-16).

(5-16) $$\frac{\partial g}{\partial y_i} - \sum_j \frac{\partial}{\partial x_j}\frac{\partial g}{\partial y_{ij}} = 0$$

6 Development of O'Brien (1970) Equations

This section uses the ideas from the previous section to shown the development of the equations from O'Brien (1970) that we need in this work, and providing some of the derivational steps which are not given in O'Brien (1970).

The analysis by O'Brien (1970) is in isobaric coordinates, which starts with the continuity equation in that coordinate system as in (6-1), where the variables have their usual interpretation: u and v are the west-east and north-south horizontal wind components, respectively, on isobaric surfaces, ω is the isobaric vertical velocity, and p is the vertical coordinate of pressure.

(6-1) $$\frac{\partial \omega}{\partial p} = -\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right]$$

Integrating (6-1) between the two pressure levels "k", and "k-1" (k increasing upwards) gives (6-2) which is the pressure-weighted divergence for one layer and is in effect a convergence with the inclusion of the minus sign but the integral will be denoted by D_k to follow the O'Brien (1970) notation.

(6-2) $$D_k = -\int_{k-1}^{k} \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right] dp = \omega_k - \omega_{k-1}$$

Applying (6-2) to a series of layers from the first level above ground (labeled as "one") to some higher level (labeled as "L") gives (6-3).

(6-3) $$\omega_L = \omega_1 + \sum_{k=1}^{k=L} D_k$$

The D_k values in (6-3) are those that are associated with a given set of model or observed analysis data. In the context of O'Brien's (1970) work, sometimes it is desired to have adjusted values of "divergence", denoted by D_k^*, which result in a specified vertical velocity target denoted by $\omega_{T,L}$, where the subscript pair "T,L" refers to a specified target at the height index "L", as in (6-4).

(6-4) $$\omega_{T,L} = \omega_1 + \sum_{k=1}^{k=L} D_k^*$$

The "f portion" of the Euler equation in (5-15) for this application as given by O'Brien (1970) is (6-5), which is a weighted sum of the squares of the differences between the original and adjusted divergences. The weighting factors κ_k are called the Gauss precision moduli, which can be defined by the error variances σ_k as in (6-6).

(6-5) $$f = \sum_{k=1}^{k=L} \kappa_k \left(D_k^* - D_k \right)^2$$

(6-6) $$\kappa_k = \frac{1}{2\sigma_k^2}$$

The constraint part of the Euler equation in (5-15) for this application starts with (6-7), which is a statement of the adjusted divergences and the specified vertical velocity target. The inclusion of the Lagrangian multiplier λ then gives the constraint equation as (6-8). Since there is only one constraint the k index is not needed.

(6-7) $$\omega_1 - \omega_{T,L} + \sum_{k=1}^{k=L} D_k^* = 0$$

(6-8) $$2\lambda\left[\omega_1 - \omega_{T,L} + \sum_{k=1}^{k=L} D_k^*\right] = 0$$

The "g function" is then giving by the sum of (6-7) and (6-8) which gives (6-9).

(6-9) $$g = \sum_{k=1}^{k=L} \kappa_k \left(D_k^* - D_k\right)^2 + 2\lambda\left[\omega_1 - \omega_{T,L} + \sum_{k=1}^{k=L} D_k^*\right]$$

Rearranging (6-9) so that all the terms which involve the vertical sum are on the left-hand side gives (6-10).

(6-10) $$g = \left[\sum_{k=1}^{k=L} \kappa_k \left(D_k^* - D_k\right)^2 + 2\,\lambda\, D_k^*\right] + 2\lambda\left[\omega_1 - \omega_{T,L}\right]$$

Applying (5-16) to equation (6-10) for a given level "N" along with substituting from (6-6) gives (6-11).

(6-11) $$\left.\frac{\partial g}{\partial D^*}\right]_{k=N} = \kappa_N 2\left(D_N^* - D_N\right) + 2\lambda = \frac{1}{\sigma_N^2}\left(D_N^* - D_N\right) + 2\lambda = 0$$

Rearranging the right-hand side of (6-11) gives (6-12), which shows the relationship between the Lagrangian multiplier, the error variance, and the difference between the original and adjusted divergences for the level N.

(6-12) $$-2\lambda\sigma_N^2 = \left(D_N^* - D_N\right)$$

In the relevant part of the O'Brien (1970) discussion, the Lagrangian multiplier is a constant so it must be a global variable for a given column. The appropriate solution is taking the vertical sum of (6-12) which gives (6-13).

(6-13) $$-2\lambda\sum_{k=1}^{k=L}\sigma_k^2 = \sum_{k=1}^{k=L}\left(D_k^* - D_k\right)$$

Solving for the term -2λ term from (6-13) gives equation (6-14).

(6-14) $$-2\lambda = \frac{\sum_{k=1}^{k=L}\left(D_k^* - D_k\right)}{\sum_{k=1}^{k=L}\sigma_k^2}$$

Substituting from (6-14) back into (6-12) gives (6-15).

(6-15) $$\frac{\sum_{k=1}^{k=L}\left(D_k^* - D_k\right)}{\sum_{k=1}^{k=L}\sigma_k^2}\sigma_N^2 = D_N^* - D_N$$

Solving for the adjusted divergence from (6-15) gives (6-16).

(6-16) $$D_N^* = D_N + \frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\sum_{k=1}^{k=L}\left(D_k^* - D_k\right) = D_N + \frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left[\sum_{k=1}^{k=L}D_k^* - \sum_{k=1}^{k=L}D_k\right]$$

Substituting from (6-3) and (6-4) into (6-16) gives equations (6-17) and (6-18).

(6-17) $$D_N^* = D_N + \frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left[\left(\omega_{T,L} - \omega_1\right) - \left(\omega_L - \omega_1\right)\right]$$

(6-18) $$D_N^* = D_N + \frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left[\omega_{T,L} - \omega_L\right]$$

Equation (6-18) is the equivalent of equation (15) in O'Brien (1970) except that there is an apparent sign error in O'Brien (1970). If the error variances are modeled as being approximately proportional to height or the height index, then (6-19) gives such an example with β being a constant.

(6-19) $$\sigma_k^2 = \beta\, k$$

Summing (6-19) up to the level "L" and using the calculus rule for a partial sum gives (6-20).

(6-20) $$\sum_{k=1}^{k=L}\sigma_k^2 = \beta\sum_{k=1}^{k=L}k = \beta\frac{L(L+1)}{2}$$

Substituting from (6-19) and (6-20) into (6-18) gives (6-21).

(6-21) $$D_N^* = D_N + \frac{\beta\, N}{\beta\frac{L(L+1)}{2}}\left[\omega_L - \omega_{T,L}\right] = D_N + \frac{2N}{L(L+1)}\left[\omega_{T,L} - \omega_L\right]$$

In O'Brien (1970) the "top level" L was at the top of the domain and the target $\omega_{T,L}$, was set to zero. In this way the net divergence was removed from the column. Thus it can be

seen that both the variational approach and the way the error variances are modeled both contribute to how the divergence adjustment is weighted in a given column.

7 Application of O'Brien Equations to Sigma-h Coordinates

The original code development for adjusting divergences to meet one or more vertical velocity targets was done in the RAMS model framework and therefore was in the sigma-h coordinate system. The sigma-h coordinate (h) is defined by (7-1), where H is the constant height above mean sea level (MSL) of the flat domain top, z is the three-dimensional MSL field of the height of the h surfaces, and E is the terrain height. The sigma-h coordinate increases with height from a value of zero at the terrain height to a value of H at the domain top. The development of the needed equations is very similar to the approach used for isobaric coordinates in the previous section.

(7-1) $$h = \frac{H(z-E)}{H-E}$$

Taking the total derivative with respect to time of (7-1) provides the relationship between the physical vertical velocity w and the sigma-h vertical velocity $\dot{h}$. In (7-2) and for the rest of this section, u and v are the west-east and north-south horizontal wind components, respectively, on the sigma-h surfaces.

(7-2) $$w = \left(\frac{H-E}{H}\right)\dot{h} + \left(1-\frac{h}{H}\right)\left(u\frac{\partial E}{\partial x} + v\frac{\partial E}{\partial y}\right)$$

Defining a height scale as in (7-3), the continuity equation in the sigma-h coordinate system is given by (7-4).

(7-3) $$M = H - E$$

(7-4) $$M\frac{\partial \rho}{\partial t} + \frac{\partial \rho u M}{\partial x} + \frac{\partial \rho v M}{\partial y} + \frac{\partial \rho \dot{h} M}{\partial h} = 0$$

Although they are included in the coded subroutines, the mapscale factors have been omitted from (7-4) and elsewhere in this section for the sake of simplicity. Integration of (7-4) between two consecutive sigma-h levels leads to (7-5).

(7-5) $$\left(\rho_k \dot{h}_k\right) - \left(\rho_{k-1}\dot{h}_{k-1}\right) = -\int_{k-1}^{k}\frac{\partial \rho}{\partial t}dh + \frac{-1}{M}\int_{k-1}^{k}\left(\frac{\partial \rho u M}{\partial x} + \frac{\partial \rho v M}{\partial y}\right)dh$$

Using the definitions given by (7-6) and(7-7) the continuity equation can be expressed in the form as given by (7-8)

(7-6) $$T_k \Delta h_k = -\int_{k-1}^{k} \frac{\partial \rho}{\partial t} dh$$

(7-7) $$D_k \Delta h_k = \frac{1}{M} \int_{k-1}^{k} \left(\frac{\partial \rho u M}{\partial x} + \frac{\partial \rho v M}{\partial y} \right) dh$$

(7-8) $$\left(\rho_k \dot{h}_k \right) - \left(\rho_{k-1} \dot{h}_{k-1} \right) = T_k \Delta h_k - D_k \Delta h_k$$

Just as in the isobaric case, if we apply (7-8) to a series of layers from the first level above ground (labeled as "one") to some higher level (labeled as "L") gives (7-9). The same approach but for adjusted divergences and a specified target gives equation (7-10). Equations (7-9) and (7-10) are the sigma-h equivalents of the isobaric forms in equations (6-3) and (6-4). The differences include a sign difference for the D_k values and the vertical coordinate increment is not part of the definition of D_k as it was for the isobaric case. The equations for the sigma-h system are presented in this way to be consistent with the software.

(7-9) $$\left(\rho_L \dot{h}_L \right) = \left(\rho_1 \dot{h}_1 \right) + \sum_{k=1}^{k=L} T_k \Delta h_k - \sum_{k=1}^{k=L} D_k \Delta h_k$$

(7-10) $$\left(\rho_L \dot{h}_{T,L} \right) = \left(\rho_1 \dot{h}_1 \right) + \sum_{k=1}^{k=L} T_k \Delta h_k - \sum_{k=1}^{k=L} D_k^* \Delta h_k$$

The equivalents of the isobaric equations (6-10) and (6-11) in the sigma-h system are given by equations (7-11) and (7-12). Again, the only differences are the extra Δh terms, the density tendency term, and the sign change on the D_k values.

(7-11) $$\begin{aligned} g = & \left[\sum_{k=1}^{k=L} \kappa_k \left(D_k^* - D_k \right)^2 \Delta h_k^2 \right] + \\ & 2\lambda \left[\left(\rho_1 \dot{h}_1 \right) - \left(\rho_L \dot{h}_{T,M} \right) + \sum_{k=1}^{k=L} T_k \Delta h_k - \sum_{k=1}^{k=L} D_k^* \Delta h_k \right] \end{aligned}$$

(7-12) $$\left. \frac{\partial g}{\partial D^*} \right]_{k=N} = \kappa_N 2 \left(D_N^* - D_N \right) \Delta h_k^2 - 2\lambda \Delta h_k = \frac{1}{\sigma_N^2} \left(D_N^* - D_N \right) \Delta h_N - 2\lambda = 0$$

Solving for the product of the Lagrangian multiplier and the error variance in (7-12) gives (7-13) which is the equivalent to the isobaric form in equation (6-12).

(7-13) $$2\lambda \sigma_N^2 = \left(D_N^* - D_N \right) \Delta h_N$$

Summing (7-13) up to level "L" and then solving for the Lagrangian multiplier term gives equations (7-14) and (7-15).

(7-14) $$2\lambda\sum_{k=1}^{k=L}\sigma_k^2=\sum_{k=1}^{k=L}\left(\overset{*}{D}_N-D_N\right)\Delta h_k$$

(7-15) $$2\lambda=\frac{\sum_{k=1}^{k=L}\left(\overset{*}{D}_N-D_N\right)\Delta h_k}{\sum_{k=1}^{k=L}\sigma_k^2}$$

Substituting (7-15) into (7-13) gives (7-16).

(7-16) $$\frac{\sum_{k=1}^{k=L}\left(\overset{*}{D}_k-D_k\right)\Delta h_k}{\sum_{k=1}^{k=L}\sigma_k^2}\sigma_N^2=\left(\overset{*}{D}_N-D_N\right)\Delta h_N$$

Solving for the adjusted divergence from (7-16) gives equation (7-17).

(7-17) $$\overset{*}{D}_N=D_N+\frac{1}{\Delta h_N}\frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left(\sum_{k=1}^{k=L}\overset{*}{D}_k\Delta h_k-\sum_{k=1}^{k=L}D_k\Delta h_k\right)$$

Substituting from (7-9) and (7-10) into equation (7-17) leads to equations (7-18) and (7-19).

(7-18)

$$\overset{*}{D}_N=D_N+\frac{1}{\Delta h_k}\frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left(\left[\rho_1\dot{h}_1-\rho_L\dot{h}_{T,L}+\sum_{k=1}^{k=L}T_k\Delta h_k\right]-\left[\rho_1\dot{h}_1-\rho_L\dot{h}_L+\sum_{k=1}^{k=L}T_k\Delta h_k\right]\right)$$

(7-19) $$\overset{*}{D}_N=D_N+\frac{1}{\Delta h_N}\frac{\sigma_N^2}{\sum_{k=1}^{k=L}\sigma_k^2}\left(\rho_L\dot{h}_L-\rho_L\dot{h}_{T,L}\right)$$

If the error variances are modeled in the same way as for the isobaric case (equations (6-19) and (6-20)) then equation (7-19) becomes (7-20), which is the equivalent of the isobaric case in equation (6-21).

(7-20) $$D_N^* = D_N + \frac{1}{\Delta h_k}\frac{2N}{L(L+1)}\left(\rho_L \dot{h}_L - \rho_L \dot{h}_{T,L} \right)$$

In our work, for a given column, equation (7-20) is actually applied twice. In the first application, the divergences are adjusted from a specified bottom level to the first target level. For the case where the model does not have clouds and the satellite data indicates clouds, this first target $\dot{h}_{T,L}$ will be positive with the target level "L" being somewhere within the observed cloud layer in question. For the case where the model has clouds and the satellite data indicates clear conditions, this first target $\dot{h}_{T,L}$ will be negative with the target level "L" being somewhere within the model cloud layer in question. In the second application, the divergences are adjusted from the previous target level upward to a specified top level. This top level usually corresponds to either the satellite observed cloud top height or the highest model cloud height, depending on which situation is at hand. The second target $\dot{h}_{T,L}$ is set to zero at the sigma level corresponding to the height z=H.

8 Divergence Adjustment Details

Multiplying equation (7-7) by the height scale M and ignoring the sigma-h increments one obtains (8-1). Defining a potential field as in (8-2) and substituting this into (8-1) one obtains the Poisson-type equation as in (8-3).

(8-1) $$M\ D_k = \frac{\partial \rho u M}{\partial x} + \frac{\partial \rho v M}{\partial y}$$

(8-2) $$\rho u M = \frac{\partial \Phi}{\partial x}, \quad \rho v M = \frac{\partial \Phi}{\partial y}$$

(8-3) $$M\ D_k = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2}$$

Equation (8-3) is solved at each sigma-h level by simultaneous overrelaxtion (SOR) with the specification of boundary conditions for the Φ field and convergence thresholds. See Press et al. 1989 for details on the SOR calculations. The Dirichlet boundary conditions set Φ values to zero on the exterior boundaries of the domain. The Neumann boundary conditions set the exterior Φ values in a way that is consistent with the wind component perpendicular to the edge in question. The Dirichlet boundary conditions have typically been used since they minimize the kinetic energy of the divergent component of the wind (Lynch 1989). The convergence threshold uses two parameters from the namelist file "**wdiag_options.inp**". These two parameters are a percentage (**u_div_per**) and a velocity scale (**u_div_scale**). The iterations stop when the mean error falls below the value given by (8-4), where H is from equation (7-1) and Δx is the horizontal mesh size in km.

(8-4) $C = 0.01 * u_div_per * u_div_scale * H * \Delta x * 1000$

The specified targets $\dot{h}_{T,L}$ may not always be realistic given their statistical origin. For example, if a target is too large near the surface then unrealistic horizontal winds can result. A parameterization has been developed which seems to remove most of these issues and is loosely based on the continuity equation. Equation (8-5) starts with the full continuity equation in the sigma-h coordinate system. Ignoring density changes, ignoring the height scale M, ignoring the difference between increments of height and sigma-h, and replacing the gradients of the horizontal wind with a horizontal wind speed S, one gets the very simple expression on the right-hand side of (8-5). This expression can be converted to a simple ratio as in (8-6). Converting this expression to a percentage (P) and adding a wind factor f_w, a height factor f_z, and the density squared results in equation (8-7).

(8-5) $$M\frac{\partial \rho}{\partial t} + \frac{\partial \rho u M}{\partial x} + \frac{\partial \rho v M}{\partial y} + \frac{\partial \rho \dot{h} M}{\partial h} \approx \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial \dot{h}}{\partial h} \approx \frac{\partial S}{\partial x} + \frac{\partial \dot{h}}{\partial z} \approx \frac{S}{\Delta x} + \frac{\Delta \dot{h}}{\Delta z} = 0$$

(8-6) $$\frac{S}{\Delta x} + \frac{\Delta \dot{h}}{\Delta z} \Rightarrow 1 = \frac{\Delta \dot{h}\, \Delta x}{\Delta z\, S}$$

(8-7) $$P = \frac{100\, \Delta \dot{h}\, \Delta x\, f_z\, f_w\, \rho^2}{\Delta z\, S}$$

If the percentage P is larger than a specified threshold, then the target $\dot{h}_{T,L}$ is iteratively reduced until P is below the specified threshold. The wind factor f_w is defined by equations (8-8) - (8-11). A minimum wind scale W_{min} which increases with height is defined using two wind parameters (S_{max} and S_{min}) which are specified in the namelist file "**wdiag_options,inp**". The wind factor f_w is then the ratio of the horizontal wind speed S and the minimum wind scale W_{min} with the limitation that f_w is less than or equal to one.

(8-8) $Z_S = z - z_1 + 1000$

(8-9) $$W_{\min} = \frac{\left(S_{\max} - S_{\min}\right)}{Z_s}\left(z - z_1\right) + S_{\min} \quad for \quad \left(z - z_1\right) \geq 1000\ m,\ otherwise\ W_{\min} = S_{\min}$$

(8-10) $$for\ \ S\ \ <\ \ W_{\min},\ f_w = \frac{S}{W_{\min}}\ and\ then\ S = W_{\min}$$

(8-11) $$for\ \ S\ \ \geq\ \ W_{\min},\ f_w = 1$$

The height factor f_z is defined by equation (8-12) and decreases with height. It is specified by the two fractions $f_{z,min}$ and $f_{z,max}$ which are specified in the namelist file "**wdiag_options,inp**".

(8-12) $$f_Z = \frac{\left(f_{z,\min} - f_{z,\max}\right)}{Z_s}\left(z - z_1\right) + f_{z,\max} \quad for\left(z - z_1\right) > 1000\ m, otherwise\ f_Z = f_{z,\max}$$

The number of iterations required to satisfy equation (8-7) is stored as a two-dimensional array and is available as an output argument in the subroutine **WDIAG**. Factors which can lead to a large number of iterations include the following: 1) one or more targets $\dot{h}_{T,L}$ were too large, 2) different parameter values which are used in calculating the wind and height factors may be needed, or 3) a different parameterization than the one in (8-7) may be needed.

9 Divergence Adjustment Calculations

9.1 Grid Considerations

One of the major choices in the **WDIAG** software is the choice of the horizontal grid to be used in the calculations. If the variable "**use_fine_grid**" in the namelist file "**wdiag_options,inp**" is set to ".false.", then the **WDIAG** calculations are performed on the original WRF Arakawa C-grid. If the variable "**use_fine_grid**" is set to ".true.", then the **WDIAG** calculations are performed on a grid which has a horizontal grid mesh size set by the variable "**dxg_fine**" in the namelist file "**wdiag_options,inp**", with a typical value being half of the original WRF Arakawa C-grid. These two choices are illustrated in Figure 9.1. The WRF Arakawa C-grid points are labeled as "+", "U", and "V" points. All of the scalars and the vertical motion are defined at the "+" points but not on the same vertical grid. The vertical motion field is on the full sigma-p grid and with the rest of the scalars on the half sigma-p grid. The horizontal u and v wind components are at the "U" and "V" points, respectively, and are on the half sigma-p grid.

If the variable "**use_fine_grid**" is set to ".false.", then in order to calculate the horizontal mass fluxes needed in equation (8-1) temperature, pressure, and water vapor must be obtained at the u and v grid points. This is accomplished by simple bi-linear interpolation. Once the horizontal mass fluxes have been calculated, the divergences are calculated at the scalar points. The adjustment of the divergences and the calculation of the potential field (equation (8-3)) are all done on the scalar grid as well. Once divergent components are calculated at the scalar points, they are then interpolated to the u and v grid locations by simple bi-linear interpolation. Because of these interpolations targets will in general not be met exactly.

If the variable "**use_fine_grid**" is set to ".true.", then all variables are defined at all points so in general the targets will be met more precisely but at the expense of

needing more CPU time to complete the SOR calculations. In this mode the user has the option to use simple bi-linear interpolation or bicubic splines to obtain the values on the fine grid.

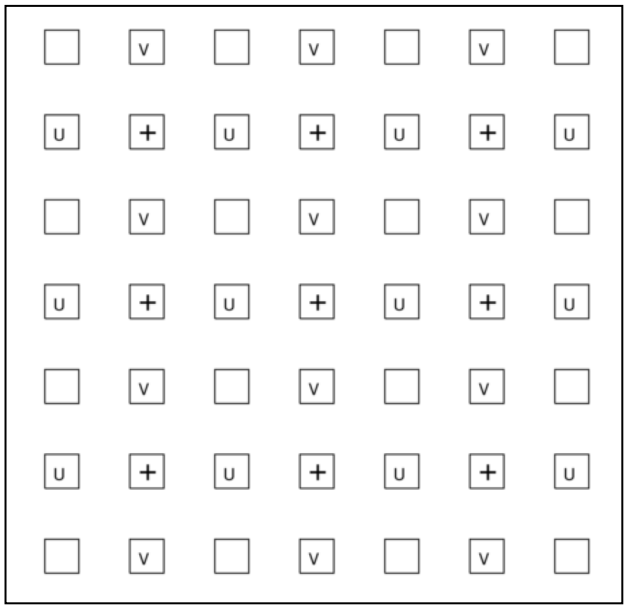


Figure 9.1 Example of WRF and WDIAG grids. The WRF Arakawa C-grid is denoted by the "+", "U", and "V" grid point locations. An example of the fine grid option in WDIAG is given by the grid point locations by the boxes and in this example has a horizontal grid size half of the original WRF horizontal grid size. See the text for additional details.

9.2 Basic Adjustment Steps

Subroutine "**do_wind_adj**" is the main driver of the divergence adjustment in program **WDIAG** and performs four basic steps which will be described briefly. The symbols for the various wind components and their meanings which will be used in the following discussion are as follows. U_O is the original WRF u wind component which is at its original grid location (**use_fine_grid** = ".false."), or has been interpolated to the fine grid (**use_fine_grid** = ".true."). In the same manner V_O is the original WRF v wind component. U_{OD} is the original (unadjusted) west-east divergent component, which for the case **use_fine_grid** = ".false.", is originally calculated at the scalar points and then interpolated back to the WRF u grid. For the case **use_fine_grid** = ".true.", U_{OD} is calculated at all the fine mesh grid points. In the same manner V_{OD} is the original (unadjusted) north-south divergent component. U_{ND} and V_{ND} are the respective differences between the original wind components and the divergent components. U_D is the new (adjusted) west-east divergent component, again either at the scalar points or the fine grid depending on the grid choice. In the same manner V_D is the new (unadjusted)

north-south divergent component. Finally, U_{NEW} and V_{NEW} are the respective sums between the U_{ND} and V_{ND} wind components and the new divergent components.

In the first step, after calculating the divergences from the original wind components, the U_{ND} and V_{ND} components are obtained as in (9-1) and (9-2).

(9-1) $$U_{ND} = U_O - U_{OD}$$

(9-2) $$V_{ND} = V_O - V_{OD}$$

In the second step, the divergences are recalculated using the U_{ND} and V_{ND} components and the vertical motion field recalculated. Apart from numerical round-off, any vertical motion at this point is the result of contributions of the density tendency term.

In the third step, the divergences are adjusted and the new divergent components U_D and V_D are obtained as well as U_{NEW} and $V_{NEW,}$ as given by (9-3) and (9-4).

(9-3) $$U_{NEW} = U_{ND} + U_D$$

(9-4) $$V_{NEW} = V_{ND} + V_D$$

In the fourth step, the divergences are recalculated with U_{NEW} and V_{NEW} and the vertical motion field recalculated. The user has the option to output U_{NEW} and V_{NEW} at their WRF u and v grid locations or at the scalar locations.

10 WDIAG Flow Charts

Figure 10.1 through Figure 10.15 provide a simplified version of the "calling tree" summary of the **WDIAG** software. Each subroutine or function is represented by a colored box. Within each box at the top the FORTRAN name of the procedure is within square brackets. Below this is a brief summary of what that procedure does. The routines have been colored coded in the following manner: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold. A call to a function or subroutine is denoted by a solid black arrow, with the tail at the routine which does the calling, and the head of arrow at the routine being called. These depictions are "simplified" to the extent that multiple calls to the same procedure are not illustrated.

Figure 10.1 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

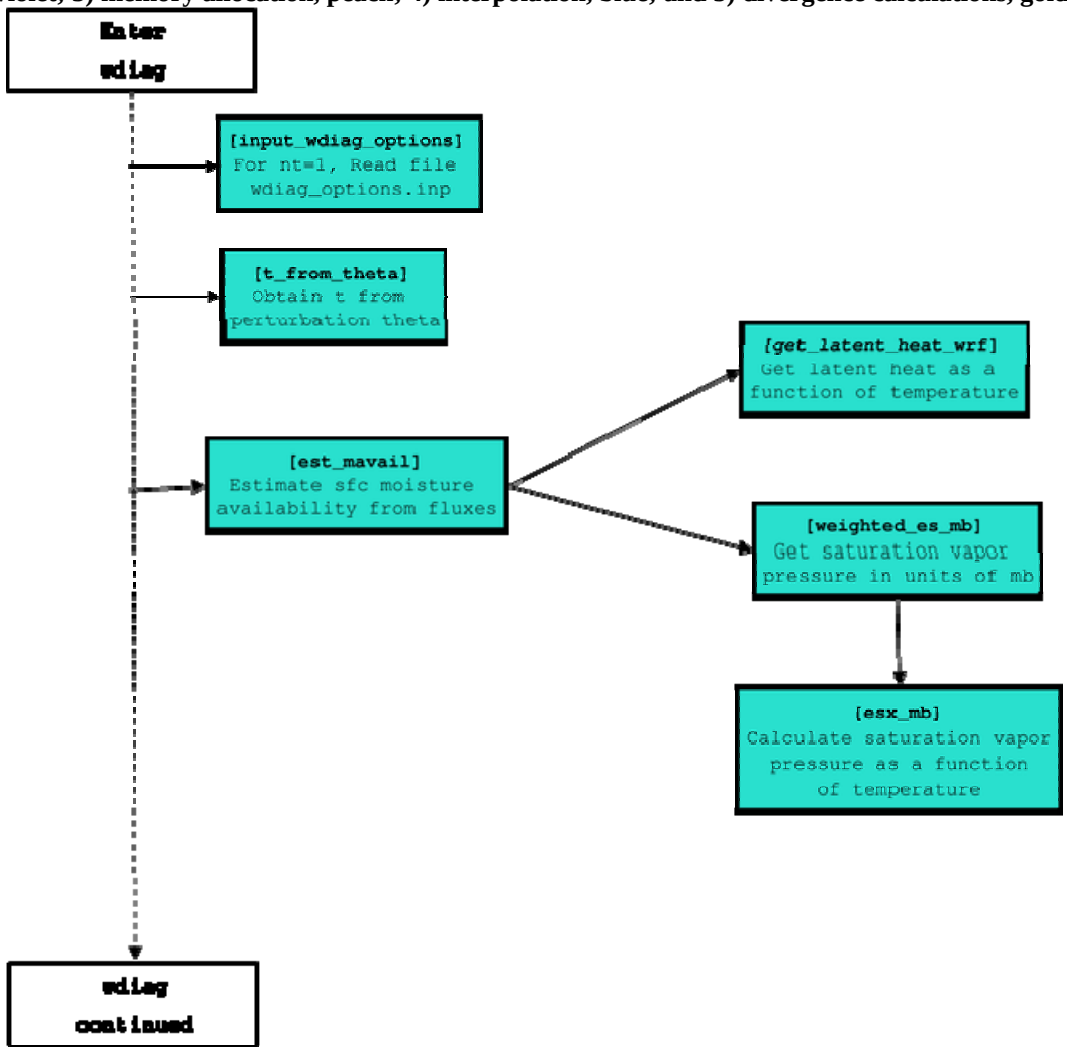


Figure 10.2 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

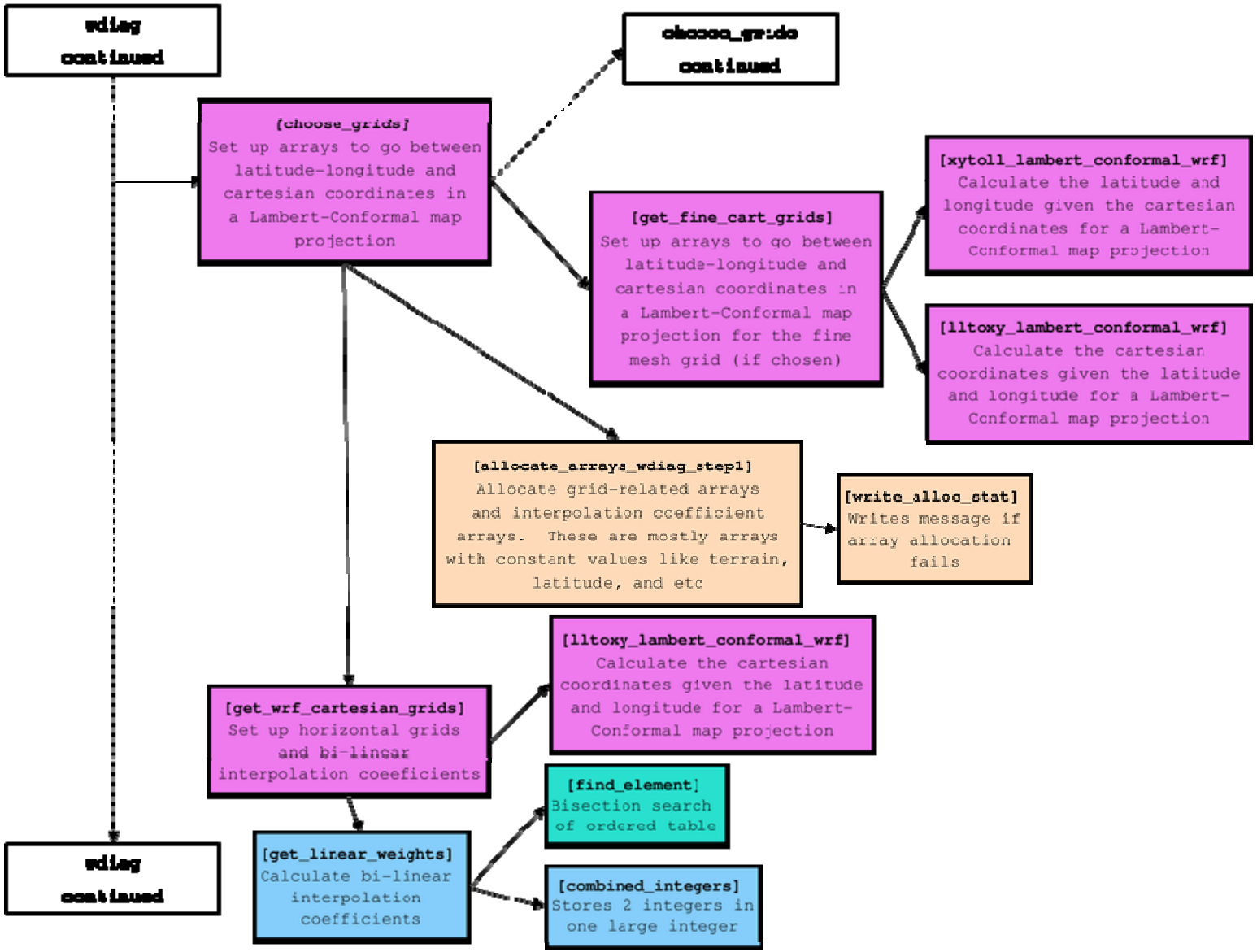


Figure 10.3 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

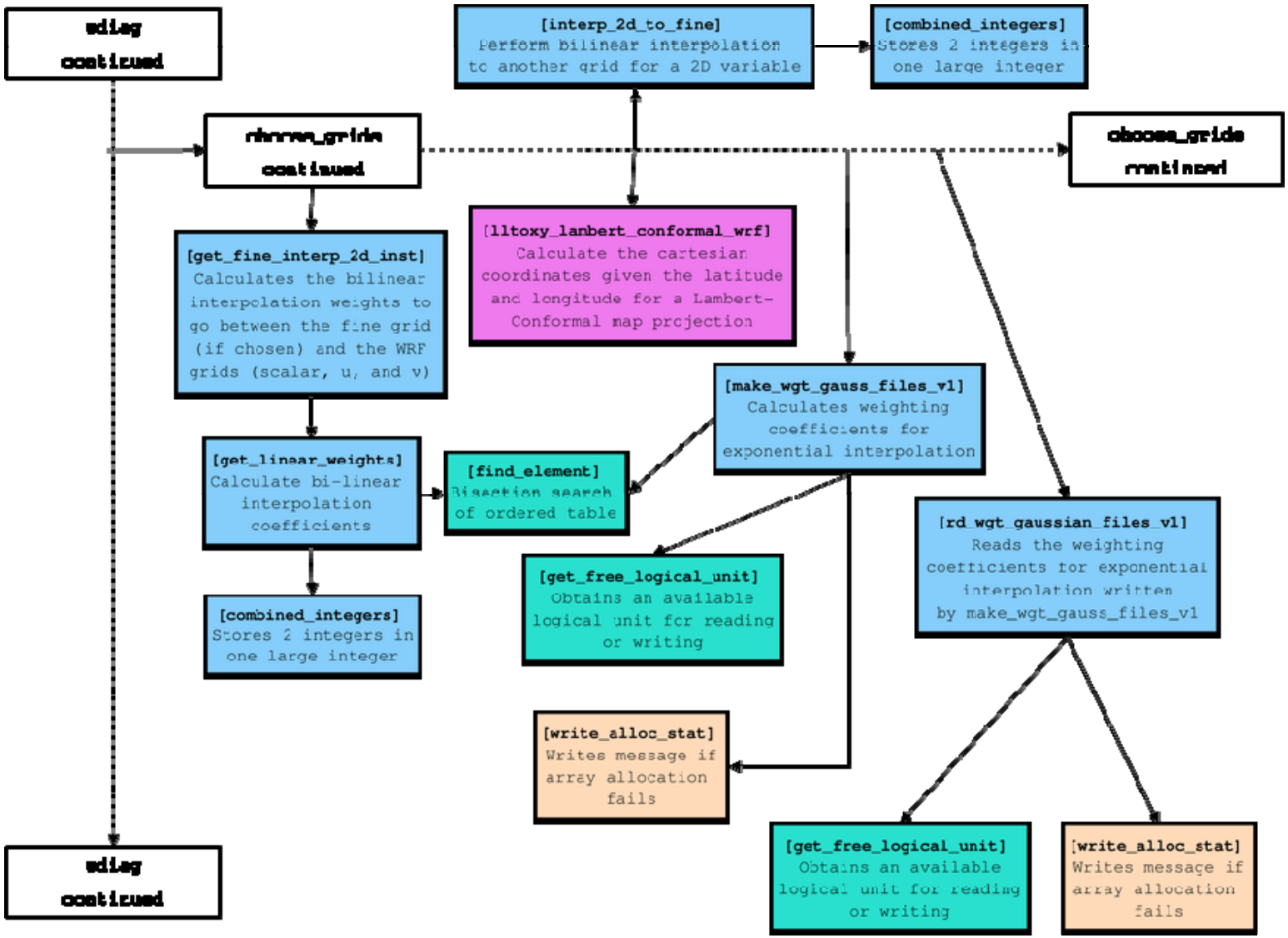


Figure 10.4 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

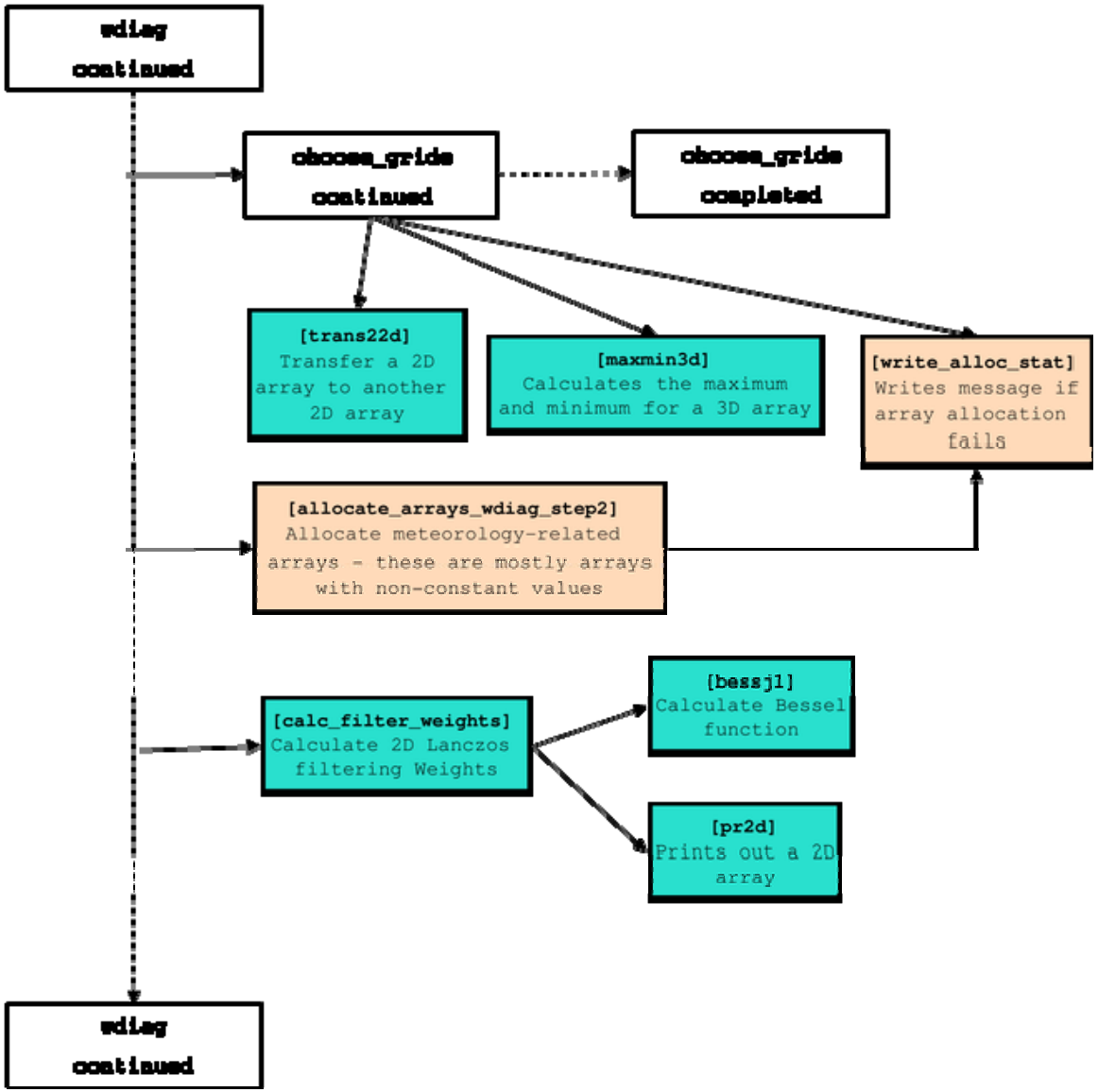


Figure 10.5 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

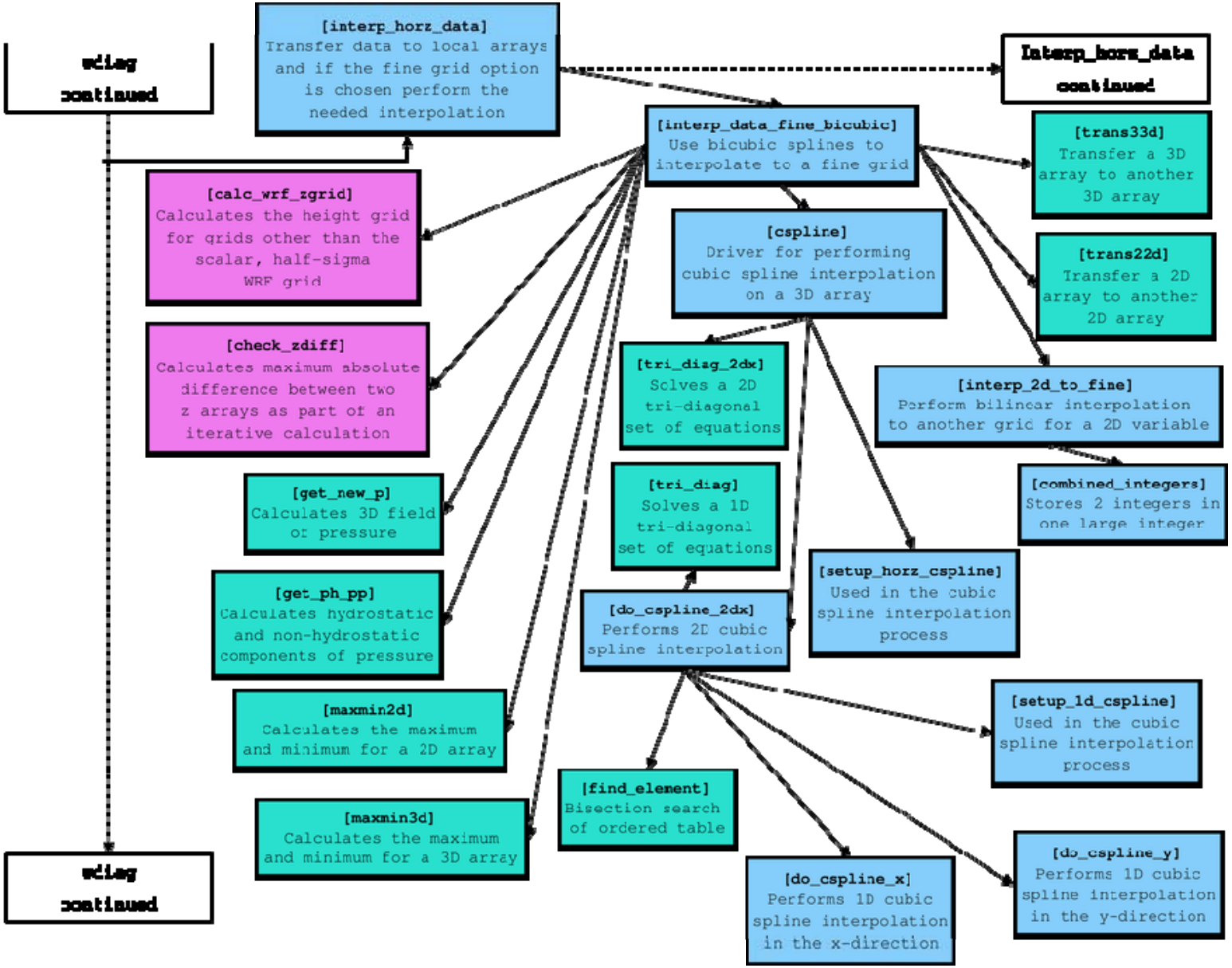


Figure 10.6 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

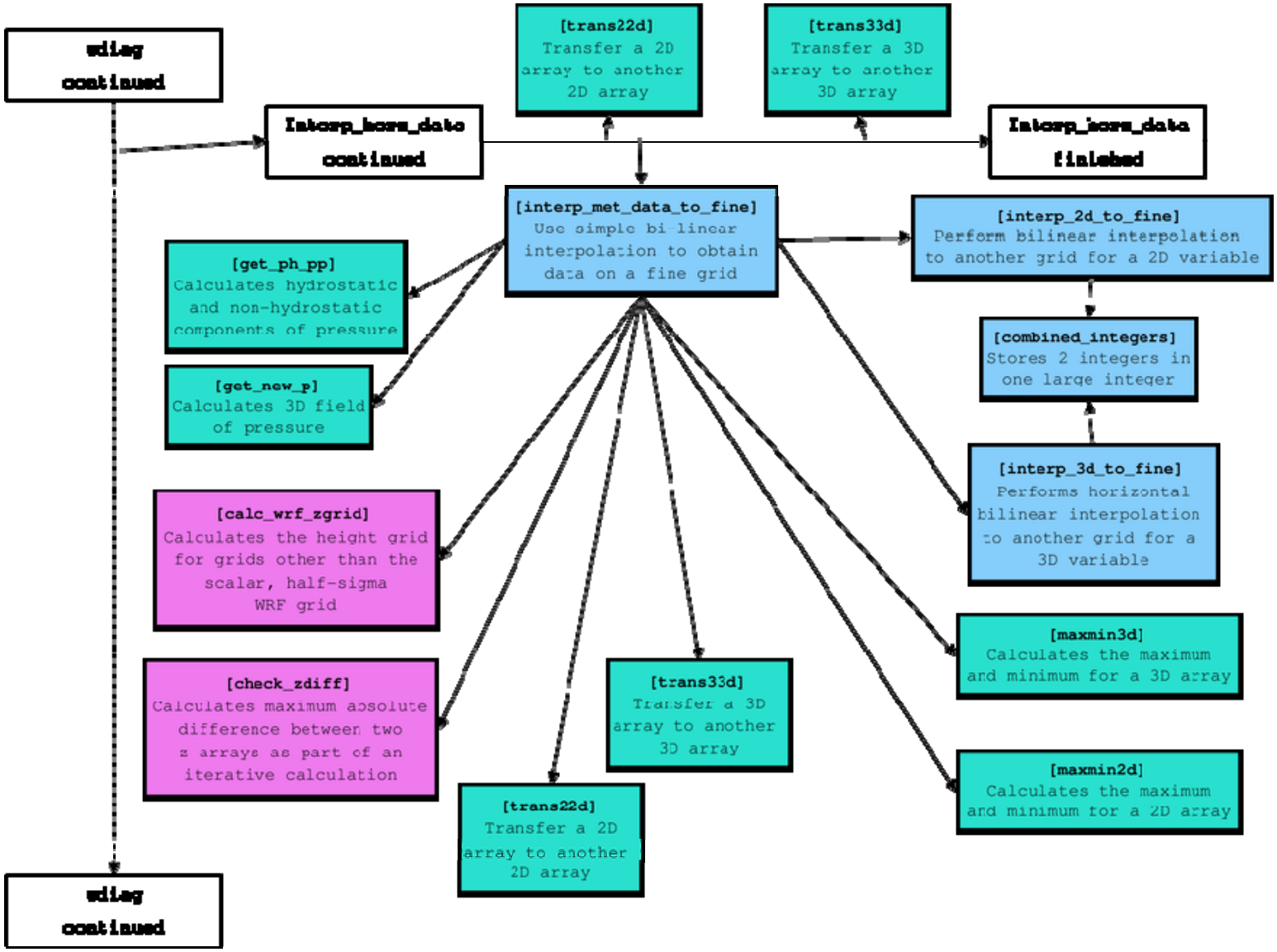


Figure 10.7 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

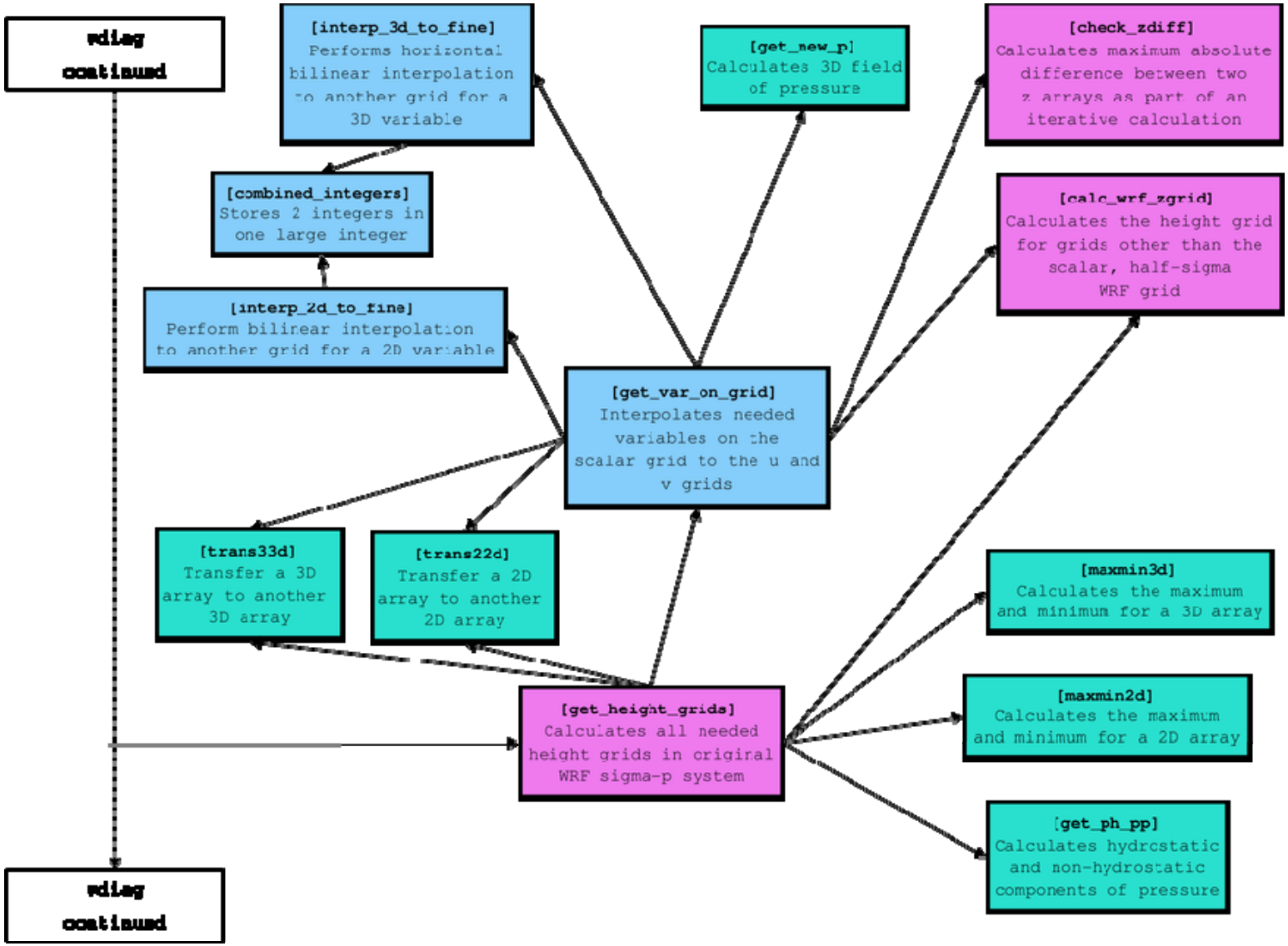


Figure 10.8 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

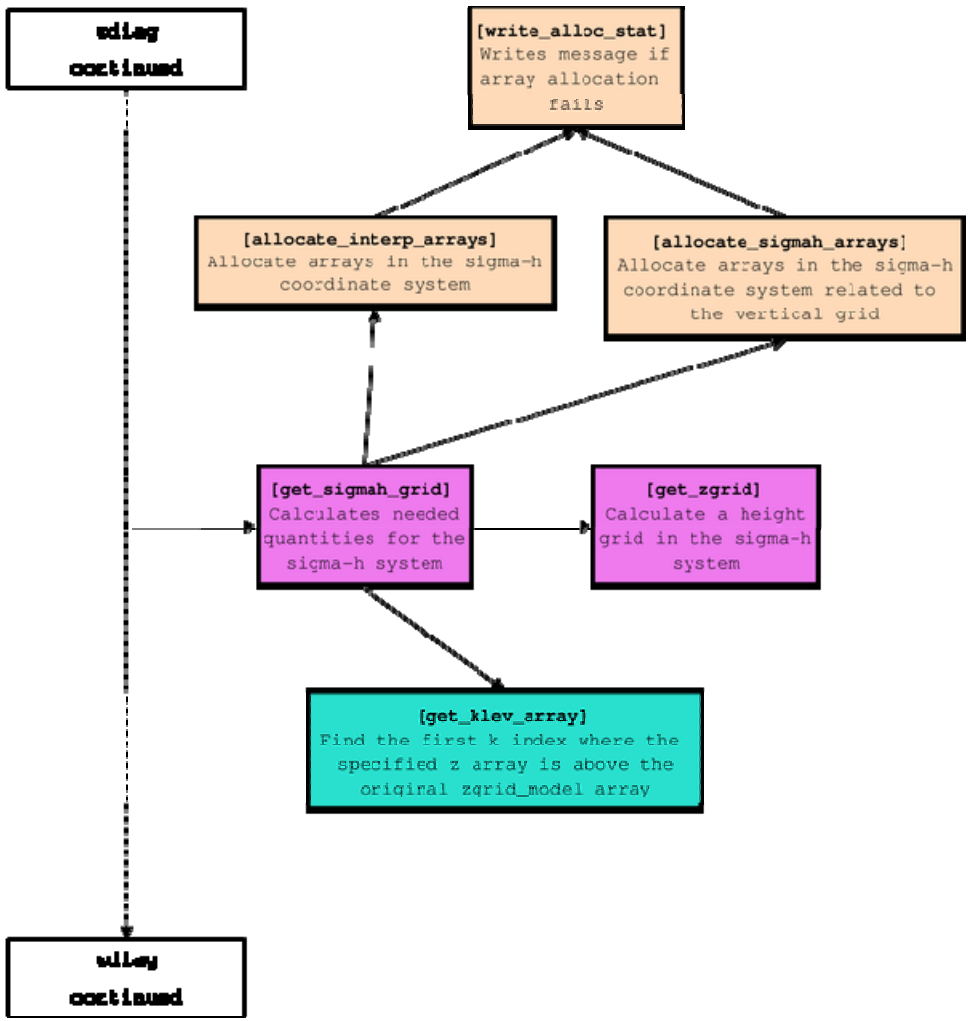


Figure 10.9 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

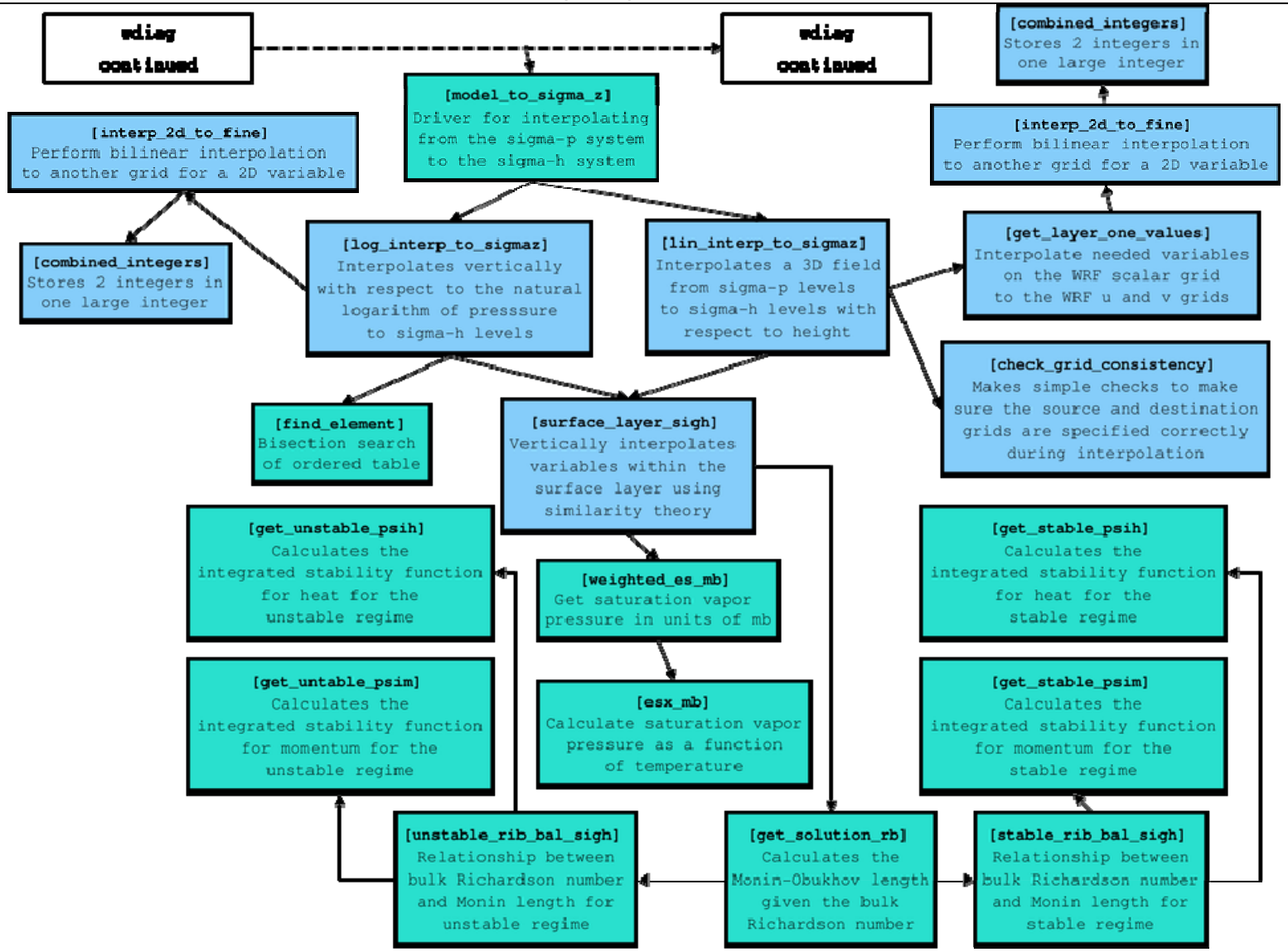

wdiag continued
wdiag continued
[combined_integers] Stores 2 integers in one large integer
[model_to_sigma_z] Driver for interpolating from the sigma-p system to the sigma-h system
[interp_2d_to_fine] Perform bilinear interpolation to another grid for a 2D variable
[interp_2d_to_fine] Perform bilinear interpolation to another grid for a 2D variable
[combined_integers] Stores 2 integers in one large integer
[log_interp_to_sigmaz] Interpolates vertically with respect to the natural logarithm of presssure to sigma-h levels
[lin_interp_to_sigmaz] Interpolates a 3D field from sigma-p levels to sigma-h levels with respect to height
[get_layer_one_values] Interpolate needed variables on the WRF scalar grid to the WRF u and v grids
[check_grid_consistency] Makes simple checks to make sure the source and destination grids are specified correctly during interpolation
[find_element] Bisection search of ordered table
[surface_layer_sigh] Vertically interpolates variables within the surface layer using similarity theory
[get_unstable_psih] Calculates the integrated stability function for heat for the unstable regime
[weighted_es_mb] Get saturation vapor pressure in units of mb
[get_stable_psih] Calculates the integrated stability function for heat for the stable regime
[get_untable_psim] Calculates the integrated stability function for momentum for the unstable regime
[esx_mb] Calculate saturation vapor pressure as a function of temperature
[get_stable_psim] Calculates the integrated stability function for momentum for the stable regime
[unstable_rib_bal_sigh] Relationship between bulk Richardson number and Monin length for unstable regime
[get_solution_rb] Calculates the Monin-Obukhov length given the bulk Richardson number
[stable_rib_bal_sigh] Relationship between bulk Richardson number and Monin length for stable regime

Figure 10.10 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

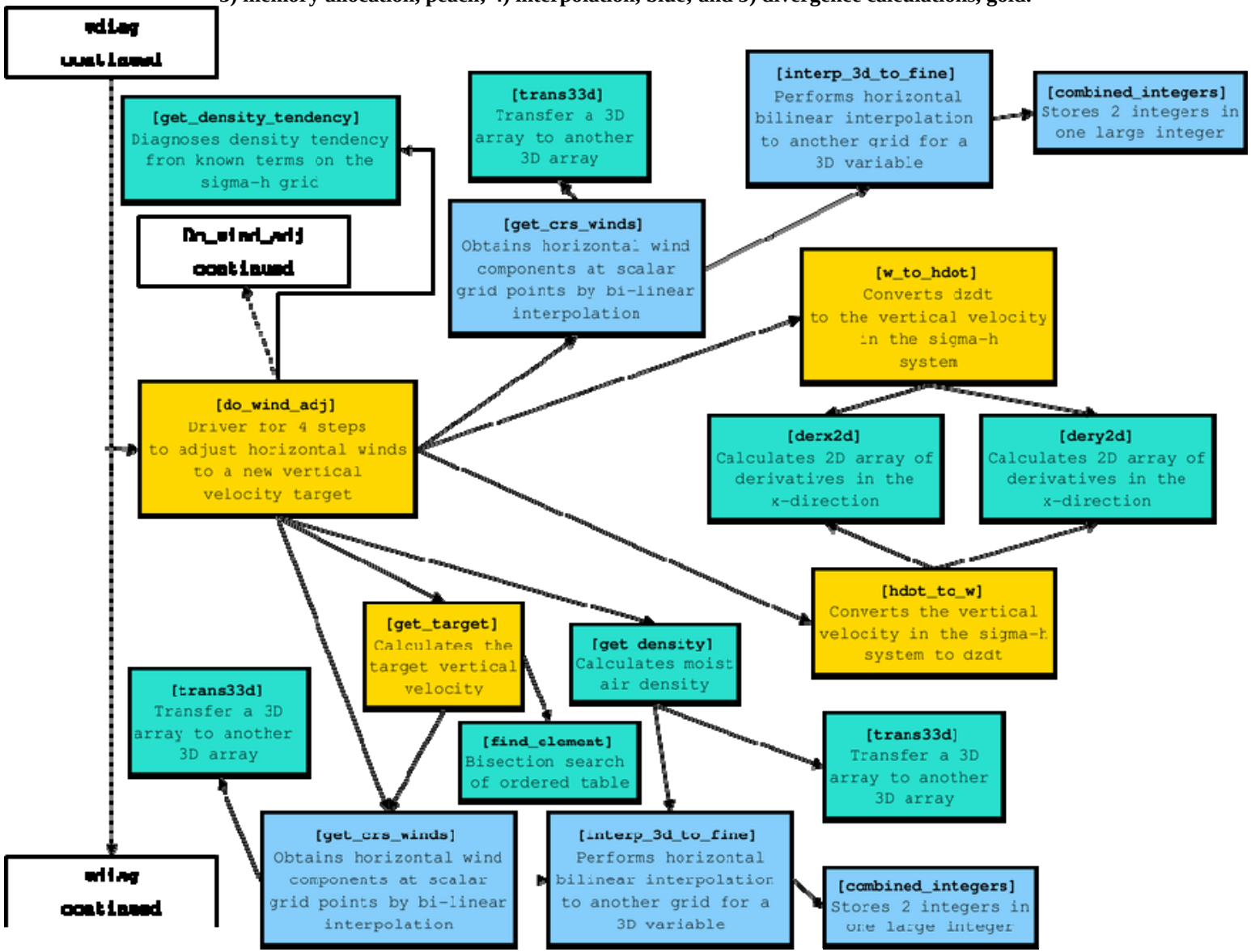


Figure 10.11 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

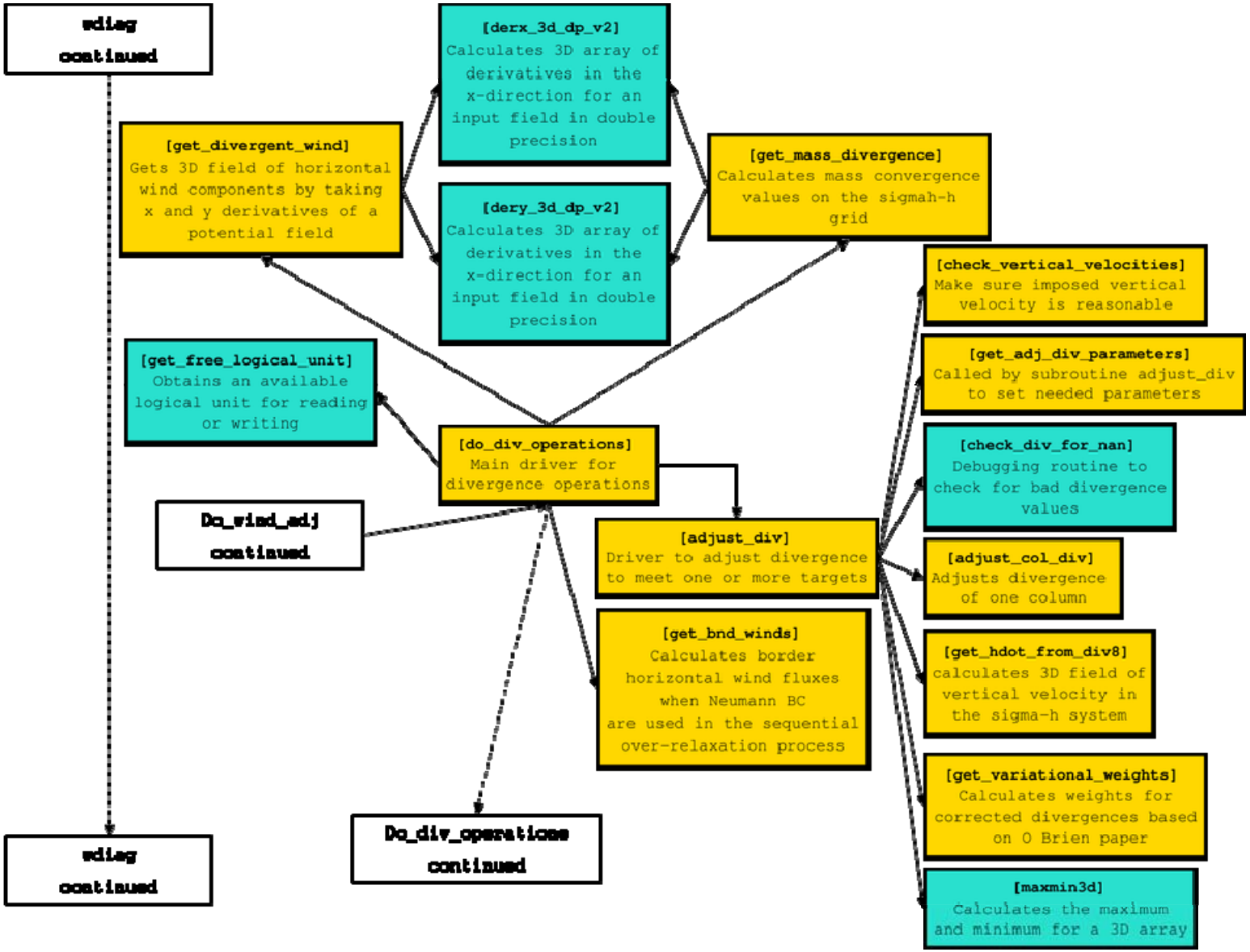


Figure 10.12 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

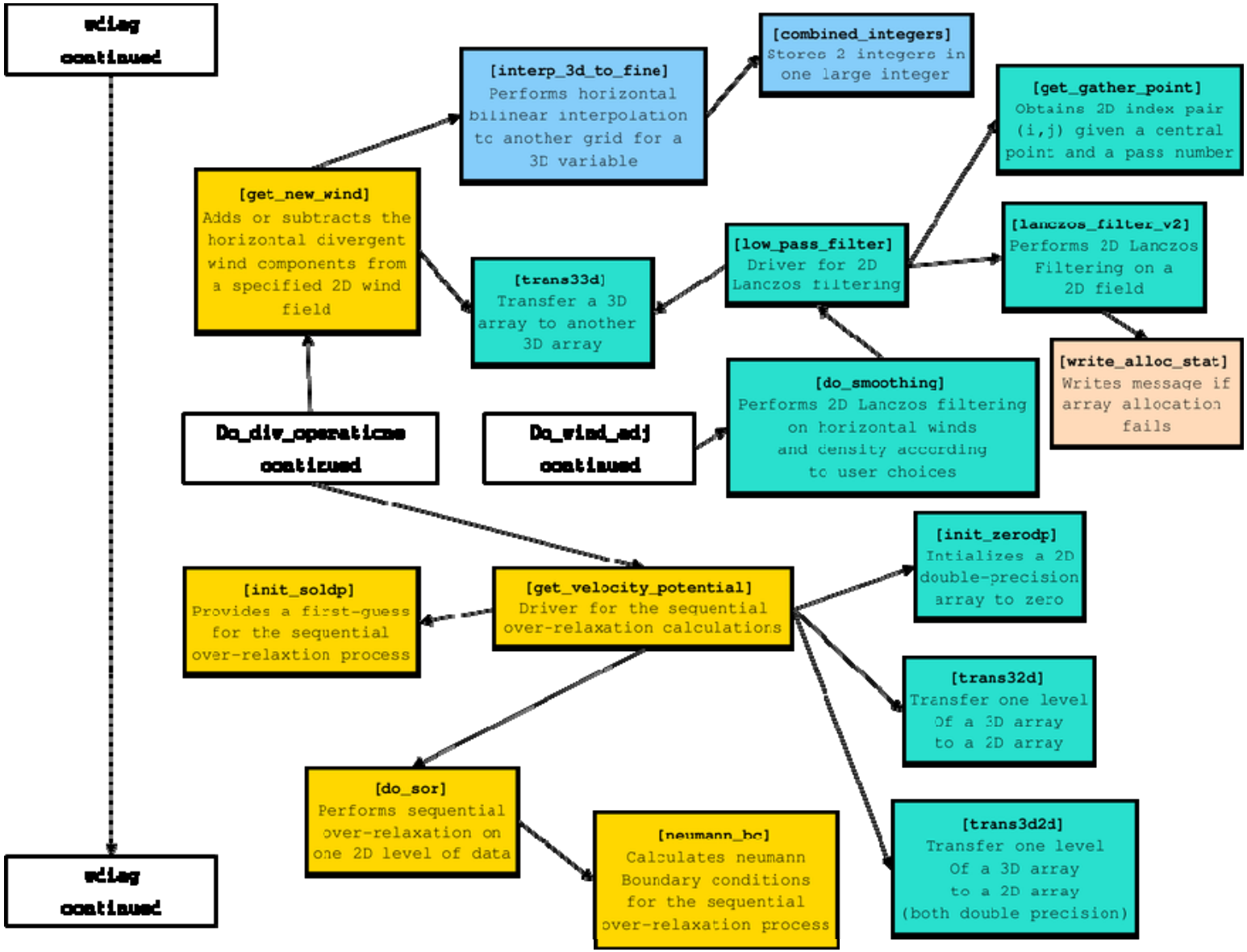


Figure 10.13 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

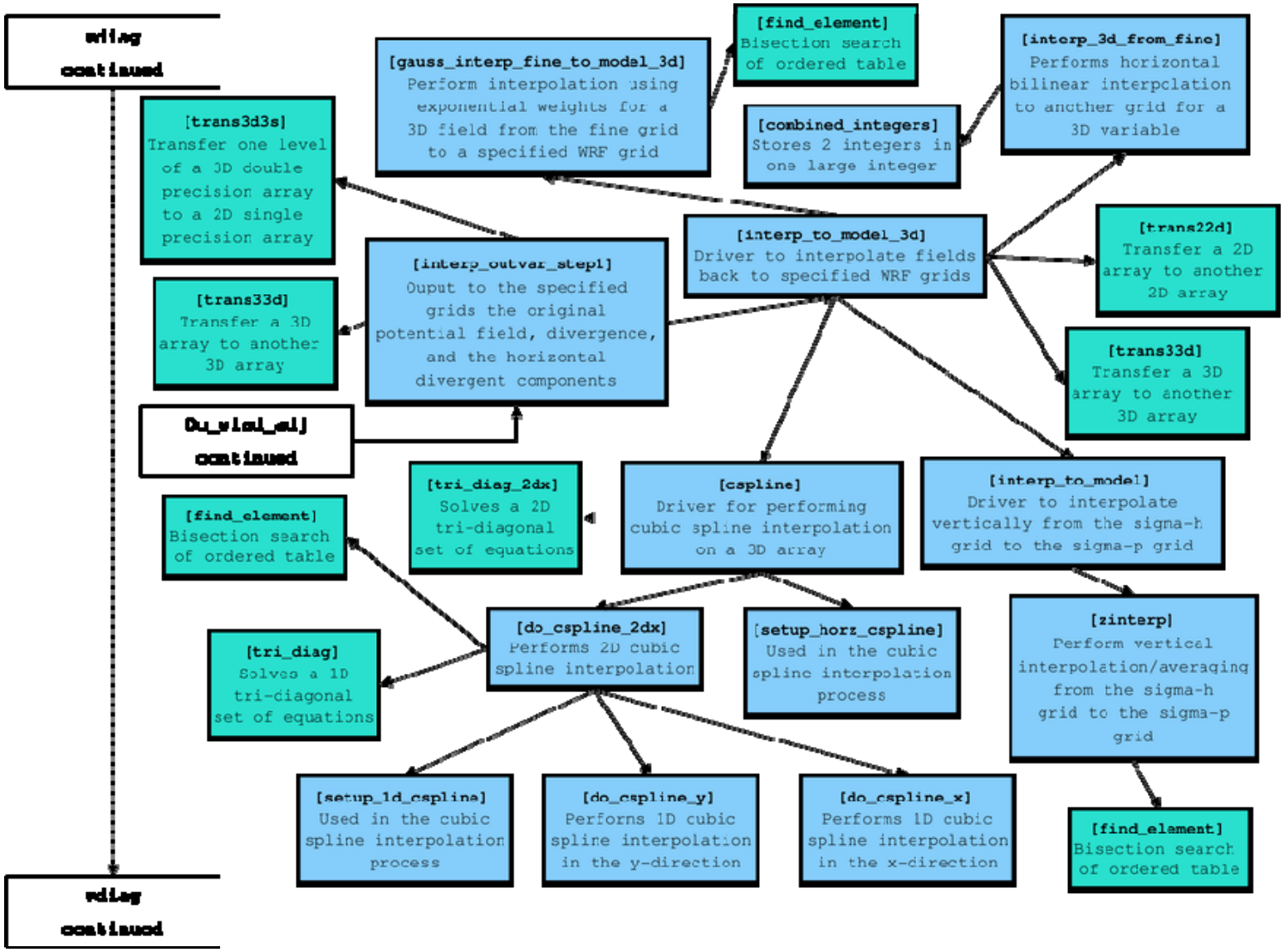


Figure 10.14 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.

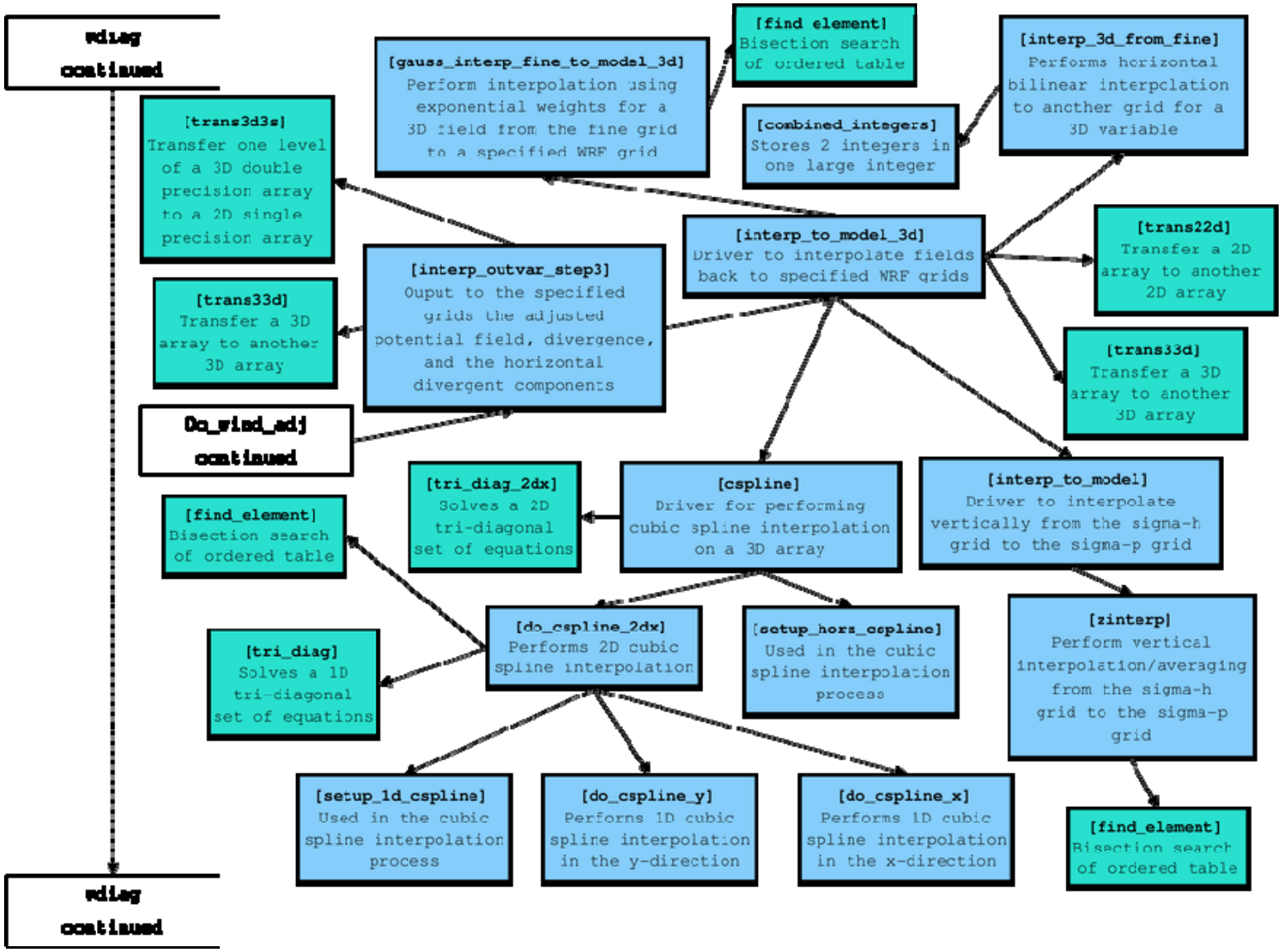
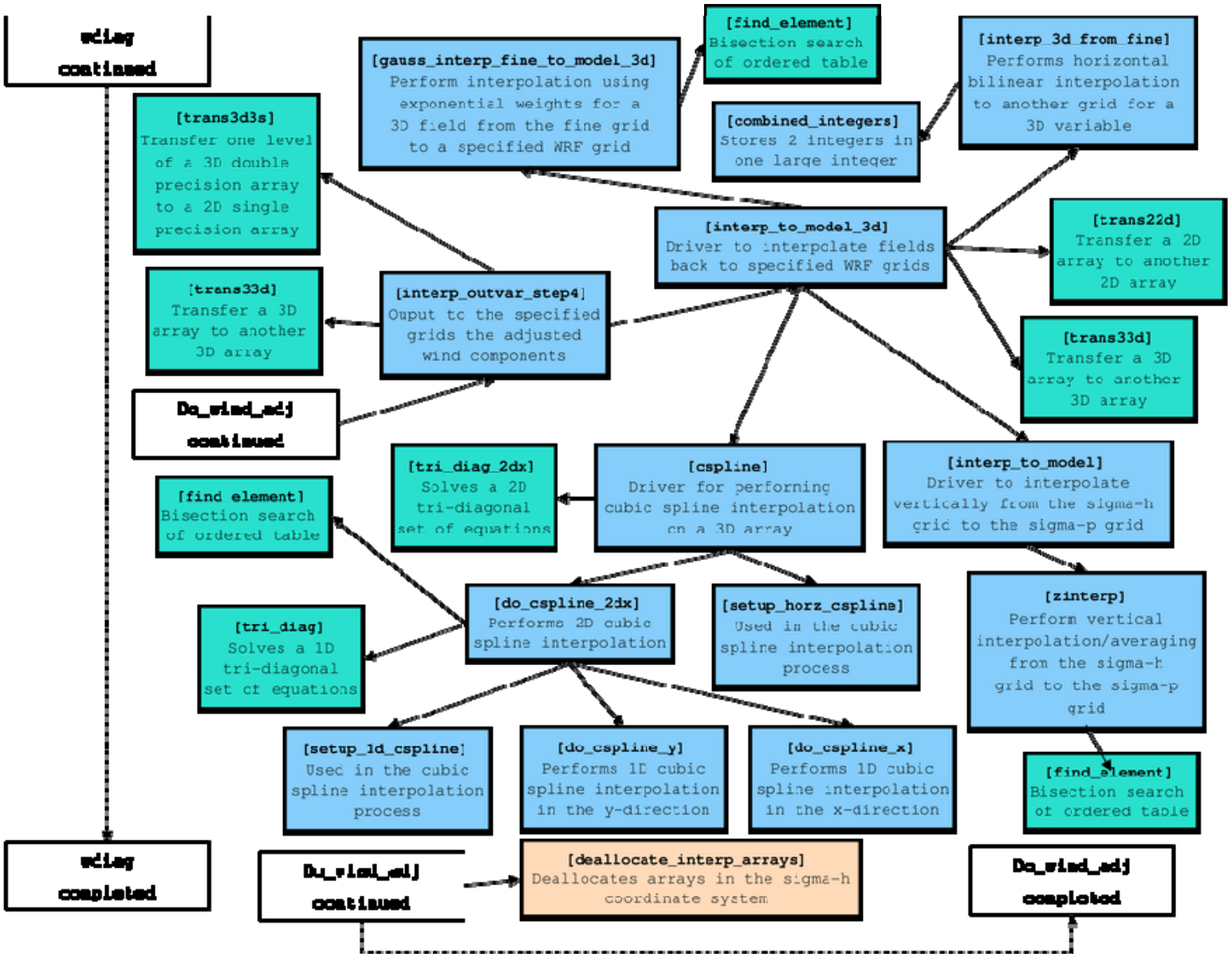


Figure 10.15 Flow chart for the WDIAG code. Routine boxes are color coded in the following way: 1) miscellaneous calculations, turquoise; 2) grid calculations, violet; 3) memory allocation, peach; 4) interpolation, blue; and 5) divergence calculations, gold.


wdiag continued
[trans3d3s] Transfer one level of a 3D double precision array to a 2D single precision array
[gauss_interp_fine_to_model_3d] Perform interpolation using exponential weights for a 3D field from the fine grid to a specified WRF grid
[find_element] Bisection search of ordered table
[combined_integers] Stores 2 integers in one large integer
[interp_3d_from_fine] Performs horizontal bilinear interpolation to another grid for a 3D variable
[interp_to_model_3d] Driver to interpolate fields back to specified WRF grids
[trans22d] Transfer a 2D array to another 2D array
[trans33d] Transfer a 3D array to another 3D array
[interp_outvar_step4] Ouput to the specified grids the adjusted wind components
[trans33d] Transfer a 3D array to another 3D array
Do_wind_adj continued
[tri_diag_2dx] Solves a 2D tri-diagonal set of equations
[cspline] Driver for performing cubic spline interpolation on a 3D array
[interp_to_model] Driver to interpolate vertically from the sigma-h grid to the sigma-p grid
[find element] Bisection search of ordered table
[do_cspline_2dx] Performs 2D cubic spline interpolation
[setup_horz_cspline] Used in the cubic spline interpolation process
[zinterp] Perform vertical interpolation/averaging from the sigma-h grid to the sigma-p grid
[tri_diag] Solves a 1D tri-diagonal set of equations
[setup_1d_cspline] Used in the cubic spline interpolation process
[do_cspline_y] Performs 1D cubic spline interpolation in the y-direction
[do_cspline_x] Performs 1D cubic spline interpolation in the x-direction
[find_element] Bisection search of ordered table
wdiag completed
Do_wind_adj continued
[deallocate_interp_arrays] Deallocates arrays in the sigma-h coordinate system
Do_wind_adj completed

11 REFERENCES

Arfken, G., 1985: *Mathematical methods for physicists*. Academic Press, Inc., 985 pp.

Duchon, . E., 1979: Lanczos filtering in one and two dimensions. *J. Appl. Met*., **18**, 1016-1022.

Lynch, P., 1989: Partitioning the wind in a limited domain. *Mon. Wea. Rev.*, **117**, 1492-1500.

O'Brien, J. J., 1970: Alternative solutions to the classical vertical velocity problem. *J. Appl. Meteor.*, **9**, 197-203.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, 1989: *Numerical Recipes (FORTRAN)*. Cambridge University Press, 702 pp.