

**Assessing High-Resolution Weather Research and
Forecasting (WRF) Forecasts Using an Object-Based
Diagnostic Evaluation**

by Gail Vaucher and John Raby

ARL-TR-6843

February 2014

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE FEB 2014	2. REPORT TYPE	3. DATES COVERED 01-10-2012 to 30-09-2013		
4. TITLE AND SUBTITLE Assessing High-Resolution Weather Research and Forecasting (WRF) Forecasts Using an Object-Based Diagnostic Evaluation		5a. CONTRACT NUMBER		
		5b. GRANT NUMBER		
		5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S)		5d. PROJECT NUMBER		
		5e. TASK NUMBER		
		5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory, Computational and Information Sciences Directorate, Battlefield Environment Division (ATTN: RDRL-CIE-M), White Sands Missile Range, NM, 88002		8. PERFORMING ORGANIZATION REPORT NUMBER		
		10. SPONSOR/MONITOR'S ACRONYM(S)		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
		12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited		
13. SUPPLEMENTARY NOTES				
14. ABSTRACT				
15. SUBJECT TERMS				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified	Same as Report (SAR)	18. NUMBER OF PAGES 94
				19a. NAME OF RESPONSIBLE PERSON

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

White Sands Missile Range, NM 88002-5501

ARL-TR-6843

February 2014

Assessing High-Resolution Weather Research and Forecasting (WRF) Forecasts Using an Object-Based Diagnostic Evaluation

Gail Vaucher and John Raby
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188		
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) February 2014		2. REPORT TYPE Final		3. DATES COVERED (From - To) October 1, 2012–September 30, 2013	
4. TITLE AND SUBTITLE Assessing High-Resolution Weather Research and Forecasting (WRF) Forecasts Using an Object-Based Diagnostic Evaluation			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Gail Vaucher and John Raby			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory Computational and Information Sciences Directorate Battlefield Environment Division (ATTN: RDRL-CIE-M) White Sands Missile Range, NM 88002-5501			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-6843		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The Model Assessment Project conducted an investigation into the applicability of the Model Evaluation Tools (MET), Method for Object-Based Diagnostic Evaluation (MODE) tool, which was designed to perform spatial verification of numerical weather prediction (NWP) model forecasts. The NWP model used during the investigation was a version of the Weather Research and Forecasting-Advanced Research Weather Research and Forecasting (WRF-ARW) model, which is tailored to address Army-scale horizontal spatial resolutions of 1–3 km. This model is called the Weather Running Estimate – Nowcast (WRE-N). The WRE-N was run over three nested grids with the 1-km inner nest grid spacing being the study focus. The observations were surface meteorological variables from independent gridded analyses. MODE compared meteorological features or “objects” defined from the forecast and observed fields for the same valid time on the basis of measureable attributes. It then quantified the differences between corresponding objects as a measure of forecast error. MODE was designed to evaluate errors in precipitation forecasts, but little work had been done with continuous variable fields. The focus of this study was to assess the application of MODE to NWP models and high-resolution meteorological variables over small, Army-relevant domains.					
15. SUBJECT TERMS object-based statistics, WRF forecast evaluations, high resolution, diagnostic evaluations, WRF-ARW, MODE, nowcasting					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Gail Vaucher
Unclassified	Unclassified	Unclassified	UU	94	19b. TELEPHONE NUMBER (Include area code) (575) 678-3237

Contents

List of Figures	v
List of Tables	vii
Acknowledgments	ix
Executive Summary	xi
1. Background	1
1.1 Goals.....	1
1.2 The Challenges	1
1.2.1 Finding an Informative Statistical Method.....	1
1.2.2 Finding a High-Resolution Observational Dataset.....	2
1.3 Evaluating Forecast Models	2
1.3.1 Weather Verification Terminology and General Approaches.....	2
1.3.2 Basic Statistics Calculations.....	4
1.3.3 Specific Verification Methods.....	4
1.3.4 Object-Based Statistics.....	6
2. MODE	7
2.1 Precipitation Analysis – the Original MODE Application.....	8
2.2 Pioneering a Continuous Variable MODE Application	8
2.3 An Overview of the MODE Process	8
2.3.1 Resolve Objects.....	9
2.3.2 Define Object Attributes	10
2.3.3 Compute Attribute Differences	11
2.3.4 Use Fuzzy Logic to Assess Total Interest Values	11
2.3.5 Statistical Summaries	12
3. Case Studies for Assessing MODE	12
3.1 Input Data for the Assessment.....	13
3.1.1 Observations.....	14
3.1.2 WRE-N Forecasts.....	14

3.2	MODE Software Execution Check	15
3.3	Calibrating MODE With a “Perfect Case”	15
3.3.1	Perfect Case – MODE Configuration.....	17
3.3.2	Six “Perfect Case” Samples	17
3.3.3	“Perfect Case” Results.....	18
3.4	A “Real World” Sample Case	22
4.	Lessons Learned	24
4.1	Improving Model and Product Assessment Efficiency	24
4.2	MODE Data Preparation – A Boundary Problem	25
4.3	MODE Data Preparation – Bracketing Data (Thresholding Values) for Finding Objects.....	25
5.	Summary and Final Comments	26
6.	References	29
	Appendix A. Example of a MODE Configuration File	31
	Appendix B. “Perfect” Case Post-Script File Output	37
	Appendix C. “Real World” Case Post-Script File Output	45
	Appendix D. MET and MODE Scripts <i>_multi</i>	53
	Appendix E. MET and MODE Script <i>run_MODE_2</i>	59
	Appendix F. MET and MODE Script <i>run_MODE_m303_2.sh</i>	61
	Appendix G. MET and MODE Script <i>run_MODE_m303_2_Perfect.sh</i>	69
	List of Symbols, Abbreviations, and Acronyms	77
	Distribution List	79

List of Figures

Figure 1. Example of MODE objects identification process on a precipitation field (NCAR, 2013).	10
Figure 2. A WRE-N three-resolution, nested-grid configuration was used for the MODE case study.	13
Figure 3. The 2013 April 23 Case Study, specifically focused on Domain 3 (1-km resolution grid), which was centered over the southern Rio Grande and Tularosa Valleys in New Mexico.	13
Figure 4. The 2013 April 23 time-series data for pressure, temperatures, relative humidity, wind speed/direction, and insolation were sampled near the center of the Case Study – WRF inner domain.	14
Figure 5. Post-Script File results summary from the Perfect Sample Case.	19
Figure 6. Post-script file results summary from the “real world” case.	23
Figure B-1. “Perfect” Case Summary page, as described in section 3.3.3.	37
Figure B-2. The “Perfect” Case Forecast raw data are graphically displayed on the top, with a color key legend on the right. The graphical display of the “Perfect” Case forecasted merged and matched objects is on the bottom of the page.	38
Figure B-3. The “Perfect” Case Observation data are graphically displayed on the top, with a color key legend on the right. The graphical display of the observed merged and matched objects is on the bottom of the page. Since a “perfect case” was used, there is no difference between figures B-2 and B-3.	39
Figure B-4. Top-diagram provides a graphical display of the “Perfect” Case forecast objects in color and the observation objects outlined in black. The bottom diagram reverses the pattern, placing the “Perfect” Case observation objects in color and “Perfect” Case forecast objects in a black outline.	40
Figure B-5. The “Perfect” Case cluster object information is displayed and tabulated. Since the case was perfect, the <i>centroid distance</i> and <i>angle differences</i> are zero grid units and degrees, respectively. The forecast and observation areas (in pixels), intersection area (in grid squares) and union area (in grid squares) are also identical magnitudes. Consequently, the symmetric differences of the objects are zero. The FCST and OBS intensity percentile variable corresponds to the intensity ratio weight variable. Total interest is 1.0000 for all clusters.	41
Figure B-6. The “Perfect” Case forecast <i>Threshold Merging</i> data are shown.	42
Figure B-7. The “perfect case” Observation <i>Threshold Merging</i> data are shown. The “perfect case” scenario generates identical merged images (bottom plot) for figures B-6 and B-7.	43
Figure C-1. The “real world” case summary page, as described in section 3.4.	45
Figure C-2. The “real world” case forecast raw data are graphically displayed on the top, with a color key legend on the right. The graphical display of the forecasted merged and matched objects is on the bottom of the page.	46

Figure C-3. The “real world” case observation data are graphically displayed on the top, with a color key legend on the right. The graphical display of the observed merged and matched objects is on the bottom of the page.47

Figure C-4. Top-diagram provides a graphical display of the “real world” case forecast objects in color and the observation objects outlined in black. The bottom diagram reverses the pattern, placing the observation objects in color and forecast objects in a black outline.....48

Figure C-5. The “real world” case cluster object information is displayed and tabulated.49

Figure C-6. The “real world” case forecast *Threshold Merging* data are shown.50

Figure C-7. The “real world” case observation *Threshold Merging* data are shown.51

List of Tables

Table 1. David Stephenson’s forecast classification scheme (<i>Types of...</i> , 2009).	3
Table 2. Dichotomous contingency table (Panofsky and Brier, 1976).	5
Table 3. Multi-category contingency table (Panofsky and Brier, 1976). The “F” is forecast and “O” is observed.	5
Table 4. Reference dates/times for case study observations and forecasts.	16
Table 5. Settings for the seven primary subdivisions of the “perfect case” and a representative “sample case.”	18
Table 6. Perfect case fuzzy engine attribute weights for calculating total Interest value.	19
Table 7. Object definition for the real world “sample case” data.	23

INTENTIONALLY LEFT BLANK.

Acknowledgments

The authors wish to thank Dr. Barbara Brown and her Model Evaluation Tool (MET) Development group at the National Center for Atmospheric Research (NCAR), for all of their constructive assistance and excellent tutorial resources provided through the initial MET/MODE learning curve and subsequent collaborations.

The authors wish to acknowledge Robert Flanigan for his support with the successful software installation and management.

Appreciation is extended to Dr. Jeffrey Smith for his iterative feedback in creating the report and Steve Kirby for his technical review.

Many thanks go to the Technical Publishing Branch at White Sands Missile Range (WSMR), NM for their consistent high standard of editing, specifically Jenny Weathers.

INTENTIONALLY LEFT BLANK.

Executive Summary

Army decisions regarding the safe and efficient employment of armed forces and equipment are impacted by the atmosphere. Advanced knowledge of atmospheric conditions can empower decision makers to increase mission effectiveness and reduce potential exposure to dangerous scenarios. The U.S. Army Research Laboratory (ARL) has been developing a tool to glean this advanced weather intelligence notification, called, the Weather Running Estimate – Nowcast (WRE-N) or Nowcast Model. Assessing and improving the “Nowcast” Model’s ability to support battlefield applications is one of ARL’s long term goals. Identifying methods for executing the high-resolution model assessments, and for finding high-resolution datasets, from which to calibrate the Nowcast Model, is nontrivial. This report documents an investigation into model verification tools, techniques, and applications relevant to assessing WRE-N.

Traditional statistical tools are useful for assessing repeatable patterns generated by lower resolution weather models, especially with the aid of largely populated data resources. This report reviews some of the standard statistical methods. However, weather conditions rarely repeat themselves exactly; data resources are often sparse; and high-resolution weather models being verified must rely on inexact equations, which are subject to spatial and temporal errors. Consequently, the traditional assessment methods have limited usefulness, forcing the assessment process to utilize nontraditional tools.

The Model Evaluation Tool (MET) package developed by the National Center for Atmospheric Research (NCAR), contains both traditional and nontraditional statistical evaluation tools. ARL’s Model Assessment Project has been conducting an investigation into the applicability of the MET package for ARL’s model Validation and Verification (V&V) mission. In particular, they have been looking into the Method for Object-Based Diagnostic Evaluation tool or “MODE,” as a resource for performing spatial verification of numerical weather prediction (NWP) model forecasts. This tool has the potential for accommodating the possible temporal and spatial mismatches that often occur in high resolution weather forecasting.

MODE compares meteorological features or “objects” defined from the forecast and observed fields for the same valid time on the basis of measureable attributes. It then quantifies the differences between corresponding objects as a measure of forecast error. The five-step process is described within the report. The statistical tool was originally designed for precipitation analyses. ARL’s applications, however, required using MODE on continuous variable applications.

Investigating the potential of this tool began by first determining if the software installation was valid. This task was assessed by running a “Perfect” case. That is, the observation and forecast data fields were identical. The results of the “Perfect” case confirmed that the software was

performing correctly. The task also was a catalyst for developing a technique that can extract data slices or intervals from continuous variable fields. Section 3 provides an overview of this technique and a detailed review of the “perfect case” results.

Running a “real world” case followed. The NWP model used was WRE-N. WRE-N is tailored to address Army scale horizontal spatial resolutions of 1–3 km. This model was run over three nested grids (9, 3, and 1 km), with the 1-km inner nested grid data being the focus of the “real world” study. Observation data from an independent gridded analysis over the same footprint was weighed against the forecast data.

To assist with the “real world” case analysis, observational time series data from a meteorological suite located near the center of the area of interest were reviewed. Distinguished atmospheric characteristics were flagged for study. While several meteorological variables were reviewed, this report presents only a sample of the results; namely, the temperature field at 2 m above ground level (AGL). The “real world” case forecast time selected was 20 Z or 1400 LT. Winds were strong, steady and westerly; implying a well mixed environment. The temperature slice extracted was: 301.00–303.00 K.

MODE results found eight simple objects in the forecast field and two in the observation field. There were 16 matching pairs, with five pairs identified as having a 70% and greater *interest* level. *Total interest* is a calculated sum based on user-defined, weighted-object attributes. The 70% *interest threshold* is also user-selected. The simple objects were all merged into one cluster for the MODE analysis. The forecast area grid squares qualifying as a clustered object, were about 25% of the area qualifying in the observation field. Based on user preferences (default settings), the Median Maximum Interest (MMI) for the Forecast was 80.42% and the Observation was 82.73%. The combined MMI (interest) was 80.42%. A visualization of the objects and the larger cluster are in appendix C.

The numerical results between the Perfect and “real world” case were very similar. However, using the visualization, the importance of the area represented in the statistics comes into focus.

Additional case studies need to be examined, before the tool can be fully optimized by the model assessment process. Exercising the numerous options in the lengthy configuration file will open up new understandings of both the tool and the V&V results. With a mastering of the MODE terms and process, this nontraditional tool has the potential to significantly strengthen ARL’s NWP model assessment process.

1. Background

The atmosphere impacts various Army decisions regarding the safe and efficient employment of armed forces and equipment. Advanced knowledge of atmospheric conditions can empower decision makers to increase mission effectiveness and reduce potential exposure to dangerous scenarios. The U.S. Army Research Laboratory (ARL) has been developing a tool to glean this advanced weather intelligence notification, called, the Weather Running Estimate – Nowcast (WRE-N) or Nowcast Model.

The numerical weather prediction (NWP) forecast model WRE-N is specifically configured for Army applications, but the strengths and weaknesses are not yet well understood. Gaining knowledge of the high-resolution WRE-N accuracy and uncertainty can improve the value of the Nowcast Model and its decision aid applications. Consequently, ARL has been researching the traditional and pioneering tools needed to accomplish high-resolution model verification.

1.1 Goals

The long term goal of this research is to assess and improve the “Nowcast” Model’s ability to support battlefield applications such as the “My Weather Impact Decision Aid” (MyWIDA). Since the quality of decisions is a function of the forecast input, assessing the “Nowcast” model was the first major milestone for this objective. The nontrivial task of identifying the method(s) for executing the assessment defined the initial target.

This report documents the investigation of model verification tools, techniques, and applications relevant to assessing WRE-N. The content focus is on the nontraditional tool called, Method for Object-Based Diagnostic Evaluation (MODE), and its use to assess the accuracy and uncertainty of the WRE-N using available gridded observation data as ground truth.

1.2 The Challenges

Progress toward generating quantitative accuracy assessments of high-resolution modeling systems, such as WRE-N, is only in its preliminary stages. The two major challenges are finding an appropriate statistical method and a correctly scaled observation data resource.

1.2.1 Finding an Informative Statistical Method

Background research confirmed that traditional statistical analyses were limited to grid-to-point techniques from which single point-forecasts were extracted at point-observational data locations. The forecast error was calculated from the difference between the forecast value and the observed value. Expanding the statistical tools to include “object” comparisons, a comparison tool for two-dimensional (2-D) model and observation “objects” was available. The tool was MODE. However, MODE had not been applied at the Army-scale. Instead, MODE was

developed for application to spatial precipitation forecasts. For this initial use, Stage II or Stage IV radar analyses of accumulated precipitation data were used for the gridded observation datasets (National Center for Atmospheric Research, 2013).

To support Army interests, MODE had to be applied to forecast scalar fields such as surface temperature, moisture, and wind at Army-relevant scales of 1–3-km resolution. This decision resulted in generating additional challenges related to the methodology for defining “objects,” which are suitable for the computation of MODE diagnostics. Fortunately, collaboration with MET developers resulted in some general “thumb rules” on the characteristics of “objects” defined from scalar fields of continuous meteorological variables.

As case study data were generated, the procedures for running MODE required modifications in order to successfully define such “objects” in the limited Army domain sizes. Further MET developers collaboration was required to achieve these modifications.

1.2.2 Finding a High-Resolution Observational Dataset

The data challenge encountered was the extremely limited availability of meteorological observation datasets for conducting model evaluations at very high resolutions.

Using the MODE tool required the availability of gridded observation ground truth datasets. The only datasets readily available were from the National Oceanic and Atmospheric Association (NOAA)/National Centers for Environmental Prediction (NCEP) Real-Time Mesoscale Analysis (RTMA) product (De Pondeca et al., 2011). The RTMA product served as the gridded observation dataset. For this work, the RTMA product, generated at a horizontal grid spacing of 2.5 km, was used.

Since the RTMA product was on a grid of 2.5-km spacing, this presented a problem because the horizontal resolution of the WRE-N forecast grid was 1 km. MODE required that the forecast and the observations grids be identical. In order to resolve this issue, the forecast grid was remapped to a 2.5-km grid spacing, to match that of the observation grid.

1.3 Evaluating Forecast Models

The decision to utilize MODE came after investigating both the standard and pioneering verification tools. In the following subsections, a summary of the standard weather statistical verification terms and techniques is provided. Section 1.3.3 will capture specific verification methods, followed by, an introduction to the nontraditional object-based diagnostic evaluation tool—MODE, in section 1.3.4.

1.3.1 Weather Verification Terminology and General Approaches

A weather forecast model predicts the future state of the atmosphere. To verify this future state, observations of what actually transpired (truth), or some good estimate, need to be assembled.

The “truth” could be either qualitative (does it look right?) or quantitative (how accurate is it?). Results from such evaluations help to monitor, improve and/or compare forecast quality.

There are many types of forecasts and verification methods. David Stephenson suggested a forecast classification scheme, which is summarized in table 1 (*Types of...*, 2009). The forecast types suggested were Forecast Nature, Space-Time Domain, and Forecast Specificity. The verification methods included visual, dichotomous, multi-category, continuous, probabilistic, spatial, and ensemble approaches. The “Forecast Specificity” type was most applicable to forecasts used in the current investigation.

Table 1. David Stephenson’s forecast classification scheme (*Types of...*, 2009).

Forecast Type	Verification Method	Example
Forecast Nature:		
Deterministic (nonprobabilistic)	Visual, dichotomous, multi-category, continuous, spatial	Quantitative precipitation forecast
Probabilistic	Visual, probabilistic, ensemble	Precipitation probability
Qualitative	Visual, dichotomous, multi-category	5-day outlook
Space-Time domain:		
Time series	Visual, dichotomous, multi-category, continuous, probabilistic	Daily max wind speed forecast
Spatial Distribution	Visual, dichotomous, multi-category, continuous, probabilistic, spatial, ensemble	Rainfall chart
Pooled Space & Time	Dichotomous, multi-category, continuous, probabilistic, ensemble	Monthly average global temp anomaly
Forecast Specificity:		
Dichotomous (yes/no)	Visual, dichotomous, probabilistic, spatial, ensemble	Fog occurrences
Multi-Category	Visual, multi-category, continuous, probabilistic, spatial, ensemble	Cold, normal, warm conditions
Continuous	Visual, continuous, probabilistic, spatial, ensemble	Maximum temperature
Object-, or event-oriented	Visual, dichotomous, multi-category, continuous, probabilistic, spatial, ensemble	Cyclone motion and intensity

Weather forecast results are generally interpreted in terms of their “goodness.” Allan Murphy (1993) defined three types of forecast “goodness:”

- Consistency, which is the degree to which the forecast corresponds to the forecaster’s best judgment about the situation;
- Quality, which is the degree to which the forecast corresponds to “truth”; and
- Value, which is the degree to which the forecast helps a decision maker to realize some benefit.

A forecast can have a high quality if it predicted the conditions that were observed, but be of little value. An example of this scenario is a forecast for clear skies over the Sahara Desert during the dry season.

Defining “truth” for a forecast verification task is generally done by using observational data. Knowing the “exact” truth can be nontrivial, since there are many uncertainties associated with atmospheric measurements. To present a detailed account of potential observational data error sources is beyond the scope of this report. Therefore, we will make the large presumption that the observation data errors are much smaller than the expected error in the forecast, allowing them to be ignored.

Verification results tend to be more trustworthy when there is a high quantity and quality of ground truth data. The need to estimate one’s confidence in the results using error bars occurs when a rare event transpires having only a small sample size, a dataset has a lot of variability, or one is investigating forecast product comparisons. A well-recognized method for establishing limits to the verification results is the confidence interval, which sets the upper and lower limits—or error bars—on the metric value, using parametric assumptions about the metric variability.

1.3.2 Basic Statistics Calculations

Basic statistical tools used in verification methods include: averages, standard deviations, mean absolute errors, bias, root mean squared error, etc. Most of these parameters are well presented in a standard college text, so they will not be reviewed here (Wilks, 2006; Panofsky and Brier, 1976; Jolliffe and Stephenson, 2012).

1.3.3 Specific Verification Methods

Standard distributions-oriented verification methods utilize approaches and plots such as histograms, box plots, scatter plots, etc. The oldest and most efficient verification method is the “eyeball” verification or “face” validation. This technique places the forecast and observation data side-by-side, allowing the knowledgeable reviewer to visually inspect for patterns and trends, and judge whether or not they appear reasonable. This approach is useful when there are only a few forecasts and quantitative verification statistics are not needed.

The Dichotomous (yes/no) forecast verification technique is another common approach. This method begins with a contingency table (see table 2). The table summarizes the frequency of “yes” and “no” forecasts, along with actually observed events. The joint distribution within the table consists of:

- Hits – the forecasted event occurred;
- Misses – the event was not forecasted, but did occur;
- False alarm – the event was forecasted, but did not occur; and,

- Correct negative – the event forecasted to not occur, did not occur.

The “marginal distribution” on the lower and right sides of the table shows the total numbers of observed and forecasted occurrences (Total Observed/Forecast Yes) and non-occurrences (Total Observed/Forecast No).

Table 2. Dichotomous contingency table (Panofsky and Brier, 1976).

		OBSERVED	OBSERVED	
		YES	NO	
FORECAST	YES	Hits	False alarms	Total Forecast Yes
FORECAST	NO	Misses	Correct negatives	Total Forecast No
		Total Observed Yes	Total Observed No	Grand total

From this contingency table, one can calculate the accuracy, bias, probability of detection, false alarm ratio, success ratio and threat score, etc. For definitions of these statistical skill scores, see Panofsky and Brier (1976).

When there are multiple categories in the weather data (e.g., bracketed temperatures, winds, or aerosol sizes), a simple approach is to use a forecast-observation histogram. The side-by-side frequency of forecast and observed categories shows how well the distributions of forecast and observations categories correspond. Distribution location, spread and skewness between forecast and observed results can be gleaned, while information on the forecast/observation correspondence is not always evident.

Another basic method is a multi-category contingency table. Table 3 is similar to the dichotomous contingency table in that it is still framed by observed and forecast data. However, the core of the table indicates the frequency of forecast and observations that fall into various predefined category bins.

Table 3. Multi-category contingency table (Panofsky and Brier, 1976). The “F” is forecast and “O” is observed.

		OBSERVED	OBSERVED	OBSERVED	OBSERVED	
	(i, j)	1	2	3	...k	
FORECAST	1	$n(F_1, O_1)$	$n(F_1, O_2)$	$n(F_1, O_3)$	$n(F_1, O_k)$	Total N(F ₁)
FORECAST	2	$n(F_2, O_1)$	$n(F_2, O_2)$	$n(F_2, O_3)$	$n(F_2, O_k)$	Total N(F ₂)
FORECAST	3	$n(F_3, O_1)$	$n(F_3, O_2)$	$n(F_3, O_3)$	$n(F_3, O_k)$	Total N(F ₃)
FORECAST	...k	$n(F_k, O_1)$	$n(F_k, O_2)$	$n(F_k, O_3)$	$n(F_k, O_k)$	Total N(F _k)
		Total N(O ₁)	Total N(O ₂)	Total N(O ₃)	Total N(O _k)	Grand Total, N

For a perfect forecast, nonzero scores would follow the diagonal, with all other entries being zero. These off-diagonal entries describe the specific nature of the forecast errors. Marginal distributions (table right and bottom) show the categorical totals for the forecast distribution when compared to the observations. Two statistical parameters that can be calculated from a multi-category data set are Accuracy and the Heidke Skill Score. For more information on these, see NCAR (2013). This table makes reducing results into a single number difficult; however, diagnosing forecast errors is easier.

For *continuous variable forecasts*, verification tools include: a scatter plot of observed and forecasted data, box plots, the calculation of mean error (bias), mean absolute error, root mean square error, mean squared error, linear error in probability space, correlation coefficient, anomaly correlation, etc.

For calculating the *probability* of an event occurring, statistical tools include: the reliability diagram, Brier score, Brier skill score, relative operating characteristic, ranked probability score, ranked probability skill score, etc.

The above methods represent the generally accepted standard approaches to verification statistics. The scientific or diagnostic verification methods that investigate the detailed nature of forecast errors tend to go beyond the distribution-oriented approaches and plots. These “nontraditional” verification methods focus on the spatial and temporal aspects of the forecast. Examples of these approaches include: neighborhood methods, scale separation methods, and spatial multi-event contingency tables.

Object-oriented methods, which focus on spatial aspects of the forecast are also on the pioneering edge of novel, informative approaches and will be the focus of the remainder of this technical report.

1.3.4 Object-Based Statistics

Object-based diagnostic evaluations answer the question, *How similar are the forecast objects to the observed objects according to a variety of descriptive criteria?* (Brown et al., 2004). They also have applications in the image processing discipline, when there is a need to condense a large amount of spatial information into a more manageable size (Jolliffe and Stephenson, 2012). The process includes identifying objects in both the forecast and observation fields of interest. Then, various characteristics of the forecast and observation object pairs are extracted, and a comparison of these objects is run. Outputs from such tools include attributes of single matched and unmatched shapes, clustered object attributes, and user-specified attributes—such as gaps between storms. Object traits can also be compared quantitatively across many cases. This multi-case approach enables investigation into systematic errors and the documentation of performance variability under different situations.

In the next section, the object-based diagnostic tool MODE will be described. Evaluating the MODE verification method was a focus in the WRE-N assessment investigation.

2. MODE

The “standard” statistical verification tools used in this study have come from the MET package, which was developed at the National Center for Atmospheric Research (NCAR), through a grant from the United States Air Force Weather Agency (AFWA). NCAR is sponsored by the United States National Science Foundation (NSF). MET is an evolving software package that began in 2008. The tools are geared for use by the NWP community, especially users and developers of the Weather Research and Forecasting (WRF) model—to assist them in assessing and evaluating model performance. These tools include the 2-D MET Point-Stat, MET Grid-Stat, and MODE. Note: MODE also has an unpublished potential for a third dimension that can be either “time” or a “vertical” orientation.

As part of the WRE-N assessment, the analyses of point-forecasts were accomplished by applying the MET subcomponent called “Point-Stat” to produce comparisons of WRF forecasts at specific points in a grid against known observations, generating statistical analyses of the differences. Point-Stat was run for a number of case-studies at the high-end of the Army-scale (1- and 3-km grid resolution), and is now configured to operate at the lower-end (as fine as 500 m resolution) assuming a representative observational dataset is available and data quality is controlled. Reports documenting the use of—and results from—cases using this tool include (Raby et al., 2012) and (Raby et al., 2011).

A lesson learned from the traditional, 2-D grid-to-point verification effort was that while the statistics from grid-to-point comparisons were helpful, assessing model performance in terms of spatial objects might better reveal the true skill of high-resolution forecasts. The V&V community already recognizes that when applying the traditional V&V techniques to high-resolution forecasts, the results often show that the errors of lower resolution forecasts are smaller than those of higher resolution forecasts. This outcome occurs because it is difficult to exactly match a forecast and observation at the same location. One cause is that the feature being forecasted may be present, but offset from the observed feature in either time or space. Another potential reason concerns how well the point observations represent the true value of the variable at a given location. A third cause can be a combination of both issues. This “double penalty” especially arises when the observations show that the observed feature was “missed” due to a nonforecast at a given location, while at the same time the forecasted feature present at another location generated a “false alarm” due to nonoccurrence of the feature. For this reason, various nontraditional techniques were developed, allowing a better skill assessment of high-resolution forecasts. These techniques, called spatial techniques, are categorized into two different types. One category, called “Neighborhood” relaxes the requirement for an exact match between the forecast and observation, through the use of a neighborhood around the forecast or observation, which influences the value at that point before error statistics are calculated. The other type is the

object-oriented—or features-based—technique, which involves a comparison of features common to the forecast and the observations (Ebert, 2008). For this study, the object-oriented type was used. The MET contains tools, which can perform both types of spatial verification. For this study, the results from the object-oriented method MODE were used.

2.1 Precipitation Analysis – the Original MODE Application

When MODE was developed by the Verification Group at the NCAR Research Applications Laboratory, the tool was designed to provide users an object-oriented spatial verification technique assessment for high-resolution forecasts. The original MODE application was to enable spatial verification of 2-D “objects” rendered from areas of precipitation as generated from both gridded meteorological observation datasets and from the model forecast grids. The two objects (a forecast and observation object) were then compared spatially (distance between their centroids, measures of shape-similarity, etc.) to quantitatively assess the accuracy of the model (National Center for Atmospheric Research, 2013). Recent work by Davis et al. (2009), of NCAR describes the use of MODE for evaluating 1-hour (h) rainfall accumulation forecasts.

2.2 Pioneering a Continuous Variable MODE Application

In this research, MODE was investigated for assessing meteorological objects rendered from gridded fields of continuous variables at the 1–3-km grid scale. This approach had not been used extensively and, although MODE was designed to allow this approach, it was discovered that there were situations where the diagnostic results used for characterizing such objects were erroneous. In addition, the object definition logic artificially limited the numbers of objects for evaluation.

Exploring the multiple functions and many potential configurations of MODE was the focus of the initial investigation. A subsequent goal will be to implement this tool in a model assessment, such as, verifying the accuracy of the WRE-N with Four-Dimensional Data Assimilation (FDDA) nudging, against available observation data. Our thesis is that applications of traditional and nontraditional techniques will provide a more complete assessment of the high-resolution WRE-N/FDDA. Our long term goal is to develop a more robust package of consolidated tools and techniques to support the future high-resolution “Nowcast” modeling requirements and developments.

In the next section, the MODE process is detailed.

2.3 An Overview of the MODE Process

MODE compares any two fields containing data from which objects may be defined. The two fields for this investigation were a gridded analysis of observations and a forecast grid. An “object” refers to regions of interest. The MODE process has five steps:

1. Define or resolve the objects.

2. Define the object attributes, such as the area, centroids, axis angle, and intensity.
3. Compute attributes differences. Examples of attribute differences include an area ratio, centroid distance, angle difference, and intensity ratios between the observation and the forecast objects.
4. Use fuzzy logic to compute the total interest values for each object pair, based on the user-defined weights. Interest quantifies how closely the forecast object matches the observed object. Match objects across fields and merge objects within the same field, based on the computed interest values.
5. Summarize the characteristics (output statistics) for the single objects, the object pairs, and the matched/merged objects.

The following summarizes the process, as documented in the MET Version 4.1 User's Guide (National Center for Atmospheric Research, 2013).

2.3.1 Resolve Objects

The process for resolving objects in a raw data field is called “convolution thresholding.” This process is done in three steps, and is demonstrated graphically in figure 1:

Step 1: Define the convolved field (C). The Convolved Field is the sum of the filter function times the raw field data.

$$C(x, y) = \sum_{u,v} \phi(u, v) * f(x-u, y-v) \quad (1)$$

where

(x,y) and (u,v) are gridded coordinates,

the filter function ϕ is a simple circular filter with radius = R and height = H. R and H are related by the requirement that the integral of ϕ over the grid is unity: $\pi R^2 H = 1$; and

the radius is the only modifiable parameter of the convolution process.

Step 2: Create a mask field, M by thresholding.

$$M(x,y) = 1, \text{ if } C(x,y) \geq T \quad (2)$$

$$M(x,y) = 0$$

otherwise, objects are the connected regions where $M = 1$.

Step 3: Restore the raw data to object interiors to obtain the object field F.

$$F(x,y) = M(x,y) * f(x,y) \quad (3)$$

Note: R (radius of influence) and T (threshold) control the entire process of resolving objects in the raw data field.

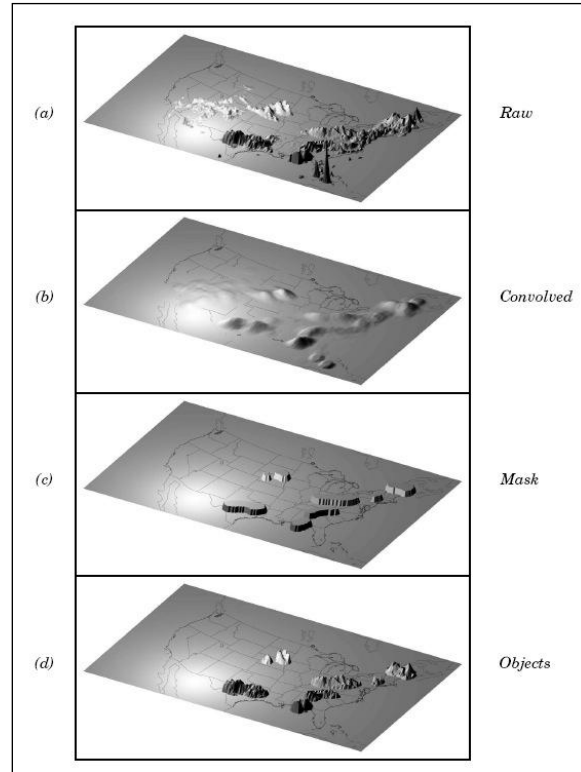


Figure 1. Example of MODE objects identification process on a precipitation field (NCAR, 2013).

2.3.2 Define Object Attributes

Object attributes are defined for single objects and object pairs. The attribute of “AREA” includes the number of grid squares that an object occupies. The “FIRST ORDER MOMENTS” (S_x and S_y) are defined as follows:

Given $\zeta(x,y) = 1$ for points (x,y) inside object, and $\zeta(x,y) = 0$ outside object; then,

$$S_x = \sum_{x,y} x\zeta(x,y) \quad (4)$$

$$S_y = \sum_{x,y} y\zeta(x,y) \quad (5)$$

The “CENTROID” is a “kind of geometric center of an object” (National Center for Atmospheric Research, 2013). This feature can be calculated from the first moments. Centroid provides a single location to a large object. An “AXIS ANGLE (θ)” gives the object an orientation and “tilt,” and is calculated from second order moments.

The “ASPECT RATIO” is defined by putting a rectangle around an object; aligning a rectangle’s axis angle to the object’s axis angle; and calculating the ratio of rectangle width to length. While the rectangle fit to the object may not be the smallest rectangle, the Aspect Ratio is calculated as the width of the rectangle divided by the length.

The “COMPLEXITY” is derived by comparing the object area to the area of its convex hull.

2.3.3 Compute Attribute Differences

Attribute differences come from comparing object pairs. The “CENTROID DIFFERENCE” is defined as the vector difference between the centroids of two objects. The “ANGLE DIFFERENCE” is the difference between the object pair axis angles. The “UNION AREA” assesses the total area that is in either one (or both) of the two objects. The “INTERSECTION AREA” is the area that is inside both objects simultaneously. And finally, the “SYMMETRIC DIFFERENCE” is the area inside at least one object, but not inside both.

2.3.4 Use Fuzzy Logic to Assess Total Interest Values

The total interest values for each object pair is determined by using fuzzy logic, based on the user-defined weights. When grouping objects together into a single field, this is called “merging.” Grouping objects together from different fields (typically the forecast and observed fields) is called “matching.”

Four MAPS are generated in this step:

1. INTEREST MAPS (I_j): Individual attributes are converted into interest values, ranging from zero (no interest) to one (high interest).
2. CONFIDENCE MAPS (C_j): Confidence Maps reflect how certain MODE developers are in the calculated attribute value. C_j ranges from zero (no confidence) to one (high confidence). This calculated value is a function of the entire attribute vector. The MODE confidence maps are generally all set to a constant value of one. The exception is the axis angle map, which uses a value of one to show low confidence and values “far from one” to indicate high confidence.
3. SCALAR WEIGHTS (W_j): Scalar Weights represent empirical judgment regarding the relative importance of the various attributes.
4. TOTAL INTEREST (T): All ingredients are collected into a single number T . T is thresholded and object pairs that have total interest values above the threshold are merged (if they are in the same field) or matched (if they are in different fields).

2.3.5 Statistical Summaries

MODE has four output files: Two ASCII files, a NetCDF file, and a PostScript file. One of the ASCII files contains contingency table statistics. This file ends in “_cts.txt”. The object and object-comparison data are in the ASCII object statistics file, which ends in “_obj.txt”. The NetCDF object file contains the raw and cluster object indices and boundary polylines for the simple objects. This file ends in “_obj.nc”. Note that the ASCII files can be read by a diversity of software packages for post-processing analysis work. NetCDF files can be viewed by using the UNIX utility “ncdump”.

The PostScript file, which ends in “.ps”, contains five pages: page one summarizes the MODE dataset application. The next two pages show the forecast and observation raw and object fields. The fourth page overlays the forecast and observation object fields. The final page contains pairwise differences for the matched clusters of objects (see the appendices B and C for examples). The results from the case studies described in the following sections were obtained from the Postscript file output only.

3. Case Studies for Assessing MODE

The high-resolution Case Study designed to investigate the possibility of using the MODE tool for WRE-N verification was defined over the southern Rio Grande and Tularosa Valleys in New Mexico (NM). The WRE-N forecast run used a three-resolution, nested grid design (9, 3, 1 km) centered on White Sands Missile Range (WSMR), NM. For this Case Study, only the data from the high-resolution, innermost domain were used. Figure 2 shows the footprint of the three domains. The footprint of the innermost, high-resolution, $127 \times 127 \times 57$, data resource is shown in figure 3.

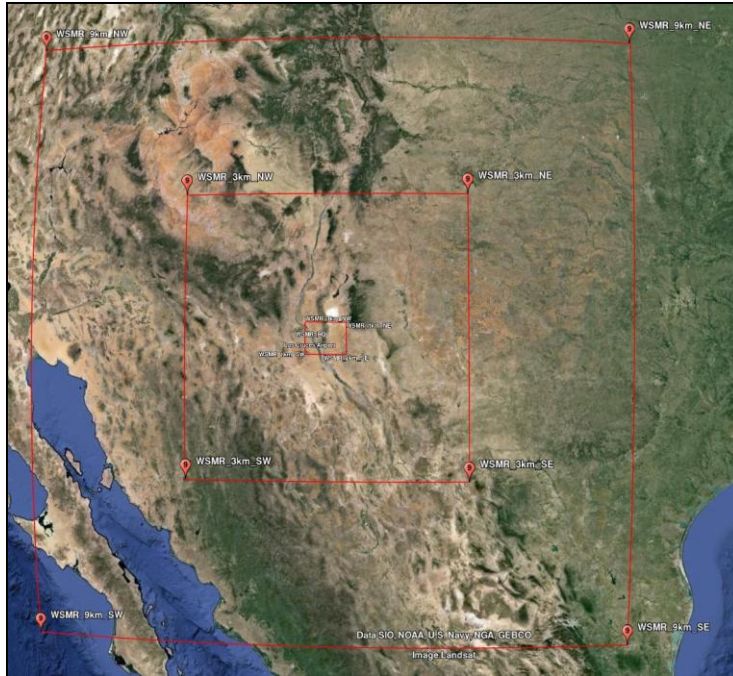


Figure 2. A WRE-N three-resolution, nested-grid configuration was used for the MODE case study.

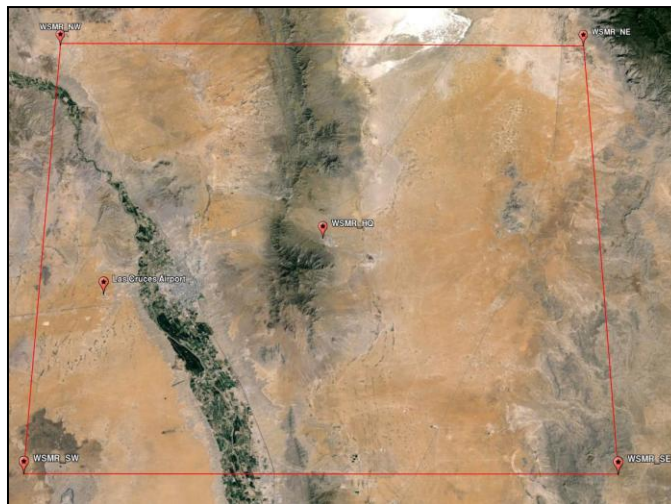


Figure 3. The 2013 April 23 Case Study, specifically focused on Domain 3 (1-km resolution grid), which was centered over the southern Rio Grande and Tularosa Valleys in New Mexico.

3.1 Input Data for the Assessment

To better grasp the general atmospheric conditions and trends for the Case Study date, point measurements were taken from a location near the center of the inner domain. The sampled variables included: Pressure (10.2-m AGL), Temperatures (10.7-, 12-, and 15.7-m AGL),

Relative Humidity (12-m AGL), Wind Speed (16-m AGL) and Direction (16-m AGL), and Insolation (12-m AGL). Figure 4 presents the results in a local midnight-to-midnight time series.

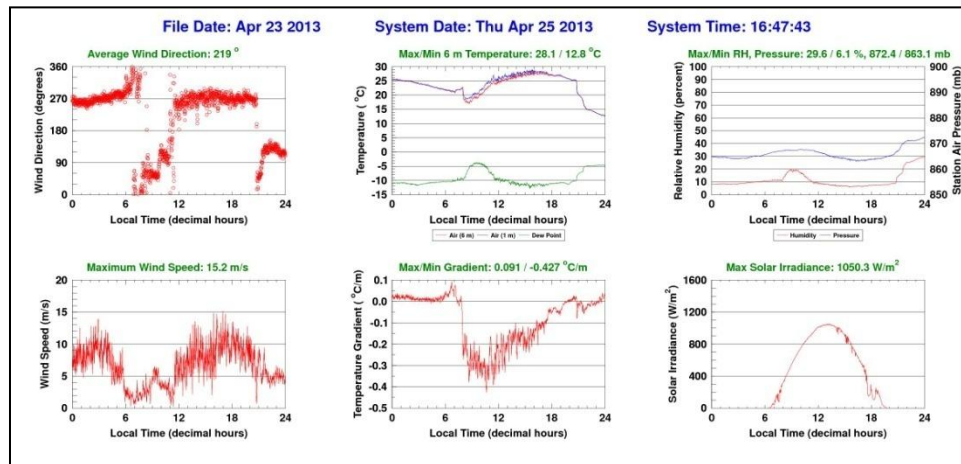


Figure 4. The 2013 April 23 time-series data for pressure, temperatures, relative humidity, wind speed/direction, and insolation were sampled near the center of the Case Study – WRF inner domain.

3.1.1 Observations

The observations used for the MODE Forecast-Observation Comparison Case Study were hourly 2.5-km resolution RTMA products. These products contained analyses of air temperature and dew point temperature at 2-m AGL, as well as wind speed, wind direction, U-wind component, and V-wind component at 10-m AGL. The data were downloaded from the NOAA Operational Model Archive and Distribution System (NOMADS). The RTMA product was generated using a 2-D variational method to assimilate point weather observations and satellite-derived measurements (National Weather Service, 2013). The products were downloaded using the NOMADS General Regularly-Distributed Information in Binary Form (GRIB) filter, which cropped the data to the WRE-N Domain 3 area centered over WSMR, NM. The files were in GRIB2 format and were converted to GRIB1 format prior to their use in the case study data generation. For this Case Study, the analyses of 2-m AGL temperature and 10-m AGL wind speed were used.

3.1.2 WRE-N Forecasts

The forecast variables used for the MODE Forecast-Observation Comparison Case Study came from postprocessed, hourly gridded forecast GRIB1 formatted output files, produced by the WRE-N model. The Case Study domain for the simulations was Domain 3, centered over WSMR, NM. The horizontal grid spacing of the model output was 1 km. Because of the mismatch of this grid with that of the RTMA product (2.5 km), the forecast grid had to be remapped to the 2.5-km grid of the RTMA data. This remapping was accomplished using the COPYGB utility, which was developed by the NOAA Environmental Modeling Center.

COPYGB performs a horizontal data interpolation to place it on the desired grid, to match the RTMA product (Developmental Testbed Center, 2013). Having a common grid for the observation and forecast datasets was a prerequisite for using the MET grid-to-grid spatial verification tool MODE.

3.2 MODE Software Execution Check

Before applying MODE as an evaluation tool, we needed to verify the successful software installation and to explore the method for identifying objects in the context of continuous variable fields.

Initial trial cases exposed two significant issues, which will be highlighted here and elaborated on in section 4. First, there was a misinterpretation of boundary data. The MODE program default labeled the border values as zero, while the statistical algorithms subsequently interpreted these zero values as “real” data. Working with the MET developers this error was fixed.

The second issue involved defining objects by extracting a subset interval within the range of the continuous data values. MODE was developed to define objects using inequality statements to determine a threshold value above or below which objects were defined. For continuous variable fields, this excludes the possibility of defining limited ranges or intervals of values that fall within some subset of the full range. Working with the MET developers, a technique was developed to define a very narrow range of the gridded data for consideration. With these two significant problems managed, the next step was to verify the software installation. This task was completed using a “perfect case.”

3.3 Calibrating MODE With a “Perfect Case”

By definition, a “perfect case” would require that all objects be consistent between the observation and forecast data. Consequently, the “perfect case” used the same RTMA gridded observation data for the “observation-truth” and the “forecast” inputs. The variable field used was Temperature at 2 m AGL. The input consisted of hourly data for 2013 April 23, from local midnight to the subsequent midnight. For convenience, the coincident Universal Time Coordinated (UTC) and local Mountain Time (LT) references for both forecast and observed data are collated in table 4. Local Time was used, to more easily interpret potential local forcing effects associated with the diurnal cycles, in the results.

Table 4. Reference dates/times for case study observations and forecasts.

Observations			Forecasts	
Date (yymmdd)	Time (UTC)		Date (yymmdd)	Time (LT)
130423	0600z	=	130423	0000 LT
130423	0700z	=	130423	0100 LT
130423	0800z	=	130423	0200 LT
130423	0900z	=	130423	0300 LT
130423	1000z	=	130423	0400 LT
130423	1100z	=	130423	0500 LT
130423	1200z	=	130423	0600 LT
130423	1300z	=	130423	0700 LT
130423	1400z	=	130423	0800 LT
130423	1500z	=	130423	0900 LT
130423	1600z	=	130423	1000 LT
130423	1700z	=	130423	1100 LT
130423	1800z	=	130423	1200 LT
130423	1900z	=	130423	1300 LT
130423	2000z	=	130423	1400 LT
130423	2100z	=	130423	1500 LT
130423	2200z	=	130423	1600 LT
130423	2300z	=	130423	1700 LT
130423	0000z	=	130423	1800 LT
130423	0100z	=	130423	1900 LT
130423	0200z	=	130423	2000 LT
130423	0300z	=	130423	2100 LT
130423	0400z	=	130423	2200 LT
130423	0500z	=	130423	2300 LT
130423	0600z	=	130423	2400 LT

3.3.1 Perfect Case – MODE Configuration

MODE specifications were defined in the MODE Configuration file. A sample of the MODE configuration file used for this case study is in appendix A. Starting with a MET tutorial configuration, the only changes made to the specifications included changing the administrative input and adjusting the three parameters, which permit one to extract a subset of the continuous variable field to define objects.

The administrative input changes included various file and variable nomenclature. For example, the user-defined model was called “WREN” and the output-prefix (output file) name was given as “m3o3WRENwsmrTMP2m_ge301p00_le303p00_PERFECT_”. The latter lengthy reference stands for “Model resolution 3 (1 km), Model domain 3 (innermost), WREN model, WSMR case, temperature data from 2 m, data extracted between greater than 301.00 K and less than 303.00 K, for the perfect case analysis”. Other administrative input included defining (1) the MET “Version” as “V4.1”, and (2) grid-res = 2.5 in (grid resolution equals 2.5 km). The field name was changed to “TMP” and “Z2” for temperature at level 2-m AGL.

To extract a subset of the continuous variable field for object definition, three attributes were adjusted:

1. Raw Thresh: The **fcst_raw_thresh** and **obs_raw_thresh** variables were used to threshold the raw fields. The default threshold set the raw fields greater than or equal to zero (so the data selected were all inclusive). For the continuous fields, this value was set to less than or equal to a user-selected number that represented the upper limit of a bracketed variable range.
2. Conv Radius: The **fcst_conv_radius** and **obs_conv_radius** variables defined the radius of the circular convolution applied to smooth the raw fields. The radii are specified in terms of grid units (see section 2.3). For the perfect cases, the convolution step was skipped by setting conv_radius to zero, thus, enabling a bracketing of variable magnitudes.
3. Conv Thresh: The **fcst_conv_thresh** and **obs_conv_thresh** variables specified the threshold values to be applied to the convolved field to define objects. The default defined objects using a convolution threshold of 5.0. For continuous fields, this value was set to greater than or equal to a user-selected number that represented the lower limit of a bracketed variable range.

After some experimentation, MODE was ready to generate some instructional output for review.

3.3.2 Six “Perfect Case” Samples

The “perfect case” was subdivided into eight subcases: Seven cases based on the default color assignment categories of the all inclusive “Perfect” Case output and one “Sample Case,” which was a realigned subset of the red color case. Table 5 summarizes the settings of the three key attributes described above. Appendix B shows a sample of the perfect case results.

Table 5. Settings for the seven primary subdivisions of the “perfect case” and a representative “sample case.”

Subcase	Conv_thresh T(low) (K)	Raw_thresh T(High) (K)	Conv_radius 0=Skip Step
Gray	>=290.22	<=292.60	0
Blue	>=292.60	<=295.66	0
Green	>=295.66	<=298.37	0
Yellow	>=298.37	<=299.73	0
Orange	>=299.73	<=300.41	0
Red	>=300.41	<=303.13	0
Sample Case	>=301.00	<=303.00	0
Pink	>=303.13	<=304.00	0

3.3.3 “Perfect Case” Results

The following description highlights the Perfect “Sample” Case results by examining the seven-page post-script (p-s) output, which is a summary of the feature-based approach used in the verification. **Note:** The results described do not include those provided in the other three MODE output files..

For the Perfect “Sample Case”, the summary table on the left side at the bottom of page 1 (see figure 5 and appendix B) lists the settings for the forecast and observation datasets in columns. The field names and model description are captured in the first seven rows. The default model reference used was “WREN,” even though for this case, the observations were used in place of the model data. The field was described as temperature (TMP) at 2-m AGL, in units of K. The initial time was 2013 April 23, 2000 UTC. The forecast validation time was the same.

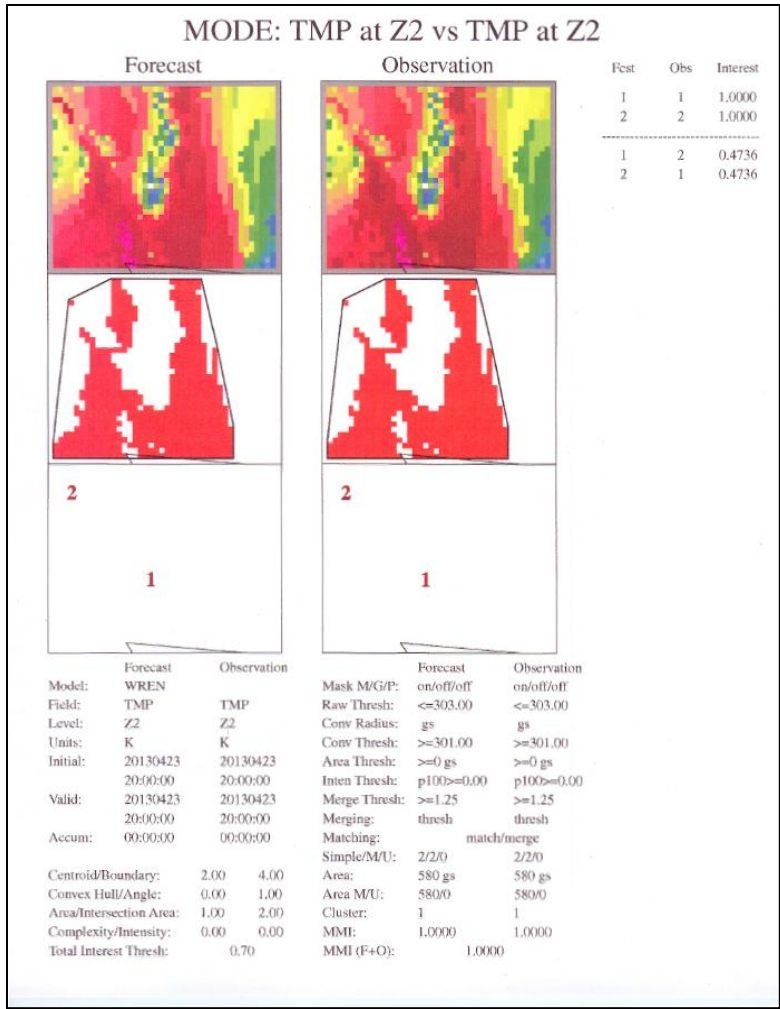


Figure 5. Post-Script File results summary from the Perfect Sample Case.

The Object Fuzzy Engine Attribute Weights included the next eight listed variables, which were used to calculate the Total Interest Value. These weights were left in their default settings and are listed in table 6. Note that these weights were not required to sum to any particular value. Rather, they are normalized by the sum of the weights, before they were used to compute the Total Interest Value.

Table 6. Perfect case fuzzy engine attribute weights for calculating total Interest value.

Variable 1/Variable 2	Variable 1 Weight	Variable 2 Weight
Centroid/boundary	2.00	4.00
Convex hull/angle	0.00	1.00
Area/intersection area	1.00	2.00
Complexity/intensity	0.00	0.00
Total interest thresh	0.70	0.70

The second column-set on this summary page (right side) are variables that define the objects. Starting at the top of the list, the mask variable deals with how missing data in the raw model and observation fields will be treated. The three mask options included Missing, Grid, and Polyline. The “Missing Mask” function takes a flagged missing data point, locates the corresponding grid point and removes this point from processing. The “Grid Mask” uses a predefined NCEP grid with which to mask the raw forecast and observation fields. The “Polyline” mask variable is the filename that defines a verification masking region. The masking regions can be specified by either a lat/lon polygon or using a gridded data file, such as the NetCDF output of the Gen-Poly-Mask tool. For the case study, the Missing Data mask was utilized (Personal Communications, Tara Jensen e-mail, November 7, 2013)

Raw Thresh, *Conv Radius*, and *Conv Thresh* were described earlier in section 3.3.1.

The *Area Thresh* specifies the area threshold values to be applied to the defined objects in units of grid squares. For this case, the area thresh variable was set to retain all objects (a default setting).

Inten Thresh specifies how the intensity values within the object are sorted. The maximum value is computed, and the intensity percentile is set to 100. Any objects with an intensity percentile that do not meet the user-selected *inten thresh* are discarded. The default setting was used in this case study, which was greater than or equal to zero.

Merge Thresh is used to define larger objects for use in merging the original objects. The thresholds are chosen to define larger objects that fully contain the original defined objects. Any two original objects contained within the same larger object are merged. For this case, the default setting of “greater than or equal to 1.25” was used.

Merging refers to grouping together objects in a single field. *Matching* refers to grouping objects together from different fields. The merging type includes four options:

1. “none” – no merging should be applied.
2. “thresh” – double thresholding merging technique should be applied.
3. “engine” – objects are merged by comparing objects to themselves using a fuzzy engine approach.
4. “both” – both thresh and engine techniques are used to merge objects.

For this case, the default double thresholding merging technique was applied on both the forecast and observation data.

The *Matching* variable also has four options:

1. “none” – no matching should be applied between fields.
2. “merge_both” – additional merging is allowed in both fields.

3. “merge_fcst” – additional merging is allowed only in the forecast field.
4. “no_merge” – no additional merging is allowed in either field. That is, each object will match at most one object in the other field.

For this case, the default of merging (“match/merge”) was allowed in both fields.

The *Simple/Merge (M)/Unmerged (U)* tallies the total object number, those objects that were merged, and those objects that were unmerged. For this case, the total number of objects was two; two objects were merged, and none were unmerged.

The area of the objects (Area) covered 580 grid-spaces for this case. All 580 were merged with zero unmerged.

A *cluster* object is any set of one or more objects in one field that match a set of one or more objects in another field. A single simple forecast object that matches a single simple observation object is also considered to be a *cluster* object (National Center for Atmospheric Research, 2013). For this case, one *cluster* was identified in both the forecast and observation fields.

The *Median Maximum Interest (MMI)* was calculated to be 1.0000 for both the Forecast and Observation fields, as well as, the two fields added together.

A list of the object-pairs by *Total Interest Level* is listed on the right side (upper) of the summary page. The highest interest is 1.0000. The dashed line was defined by the user-selected *Total Interest Thresh*, which was listed under the Object Attribute Weights section (lower left). The pairs above the dashed line are considered interesting enough for further processing.

In the “Perfect Sample Case,” there was no surprise that the two objects identified exceeded the 70% *Total Threshold*. Each of these objects was identified numerically (1 and 2) in the bottom forecast and observation graphical displays. The middle graphic showed the 1 *cluster*. The Matching color across the two fields (forecast and observation) indicated matching objects. The black perimeter outline indicated the use of merging. These features concurred with the perfect case’s “Matching” setting of “match/merge.”

On page 1 of the p-s output, the top graphic showed the raw data values for both Forecast and Observation. The identical nature of the forecast and observation plots confirmed the perfection of the Case being studied.

The subsequent pages of this summary document enlarge the material presented on the first page. For example, on page 2 (see figure B-2), the Forecast raw data were graphically displayed on the top, with a color key legend on the right. The graphical display of the forecasted merged and matched objects was on the bottom of the page.

Page 3 of the p-s output (see figure B-3) followed the same pattern only with the Observation data. Since a perfect case was used, there was no difference between page 2 and 3.

Page 4 of the p-s output (see figure B-4) top-diagram provided a graphical display of the forecast objects in color and the observation objects outlined in black. The bottom diagram reverses the pattern, placing the observation objects in color and forecast objects in a black outline.

The cluster object information was displayed and tabulated on page 5 of the p-s output (see figure B-5). Since the case was perfect, the *centroid distance* and *angle differences* were zero grid units and degrees, respectively. The forecast and observation areas (in pixels), intersection area (in grid squares) and union area (in grid squares) were also identical magnitudes. Consequently, the symmetric differences of the objects were zero.

The “FCST” (forecast) and “OBS” (observation) intensity percentile variable corresponds to the intensity ratio weight variable. In the table, both the 50% (*int50*)—or median value—and 90% (*int90*) were listed. The only cluster cited had 580 grid-spaces. Since the same data were used for forecast and observations, the two 50% thresholds (median) were 301.66 K. The two 90% intensities were calculated as 302.56 K. Total interest was 1.0000 for all clusters.

The last two pages of the p-s summary (see figures B-6 and B-7) showed the *Threshold Merging* for the forecast and observation data. The perfect case scenario generated identical images. As expected, the intended threshold map on the bottom covered just the red color range used in the composite image at the top. These diagrams confirmed that the intended variable range was selected in this analysis.

The perfect case results confirmed that the MODE software had been installed and was operating correctly. The next step was to exercise MODE on a “real world” model forecast with observations.

3.4 A “Real World” Sample Case

The date for the “real world” case was April 23, 2013 (same as the “perfect case”). The MODE configuration file began with the MET tutorial (default) configuration file. The only changes made to the specifications included changing the administrative input and adjusting the three parameters, which permitted one to extract a subset of the continuous variable field to define objects. This was necessitated by the input of “real world” forecast data.

The administrative changes included defining the forecast model as the WREN Model (this time for real). The forecast versus observation fields examined included Temperature (2 m), Dewpoint (2 m), Wind (10 m), and U-grid (10 m). For this report, only the Temperature results at 2-m AGL will be discussed.

The three parameters changed to extract only the “sample case” data from the “real world” resources were *Conv_thresh*, *Raw_thresh*, and *Conv_radius*. The values chosen were to focus specifically on the subcase consisting of temperature values that were assigned a red color. The values used are listed in table 7.

Table 7. Object definition for the real world “sample case” data.

“Red” Subcase	Conv_thresh T(low) (K)	Raw_thresh T(High) (K)	Conv_radius 0=Skip Step
Sample Case	>=301.00	<=303.00	0 (gs)

The p-s file summary confirmed the above changes (see figure 6 and appendix C) and documented the “real world” case specifications. Such specifications included: The initial forecast date and time of April 23, 2013, 0600 Z. The forecast valid time was 2000 Z. The initial and valid observation data were from April 23, 2013, 2000 Z. The 2000 Z, or 1400 LT, time period was selected, after reviewing the time series explained in section 3.1 and several of the April 23rd Model-Observation results. The clarity of contrast with the perfect case was one of the primary purposes for choosing this time period.

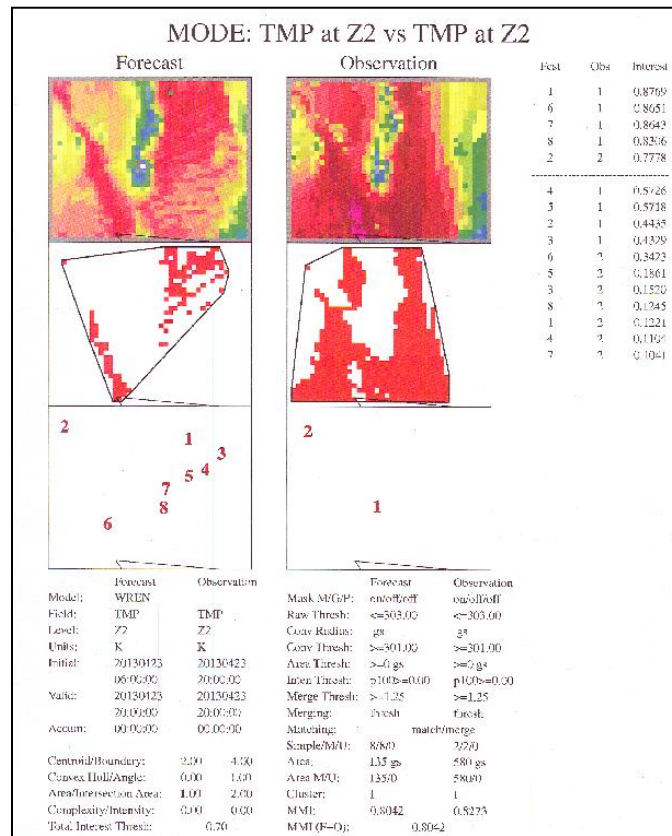


Figure 6. Post-script file results summary from the “real world” case.

The weights used to calculate the total interest, were aligned with the “perfect case” (see table 6: the perfect case; figure 6: “real world” case p-s file). The “real world” variables were also the same as the “perfect” case.

From the p-s output, the “real world” results identified eight forecast objects and two observed objects. Eight forecast objects were merged, and zero were unmerged. The two observation objects were merged and zero unmerged. The total forecast area reported consisted of 135 grid-squares, with 135 merged and zero unmerged. The observation results covered a much larger area with 580 grid-squares; 580 observed data grid-squares were merged and zero unmerged.

One cluster was reported for both the forecast and observation resources. The MMI was about equal at 0.8042 (fcst) and 0.8273 (obs). The combined MMI from the set of all interest values from the forecast and observed objects was 0.8042.

The *Total Interest Thresh* designated five Forecast/Observation pairs that had an interest above the 70% threshold. The other 11 pairs fell below this threshold line.

The forecast and observation results showed that both extracted fields were clustered together into single objects. The close proximity and geometric shape of these objects and their pixels were shown on the “Forecast Objects with Observation Outlines” and “Observation Objects with Forecast Outlines” plots of the postscript file (see figure C-4).

Reviewing the Cluster Object information (p-s output, page 5, figure C-5), the cluster pair was described as having a centroid distance separation of 9.93 grid-squares. The angle difference was 25.60°. The forecast object area consisted of about 25% of the grid-squares of the observation object area. The area that was intercepted was 72 grid-squares—or about 11% of the total area being considered (union is 643 grid-squares). The symmetric difference was 571, indicating there was limited common attributes between the objects.

The median point for the forecast (301.16 K) was a tiny bit cooler than the observation median (601.66 K). The 90% intensity threshold for the forecast and observation fields was 1.15 K, with the forecasting showing a cooler threshold. The total interest for this cluster was 0.8979—or about 10% less than a Perfect fit.

4. Lessons Learned

4.1 Improving Model and Product Assessment Efficiency

In 2011, the Model and Product Assessment Project Group developed and published a process to automate the use of the MET. A set of scripts facilitated running the grid-to-point verification code, as well as the process for verifying a gridded forecast field against point ground truth observations. In preparing for the MODE assessment, a need arose for a more efficient execution of the MET statistical analyses. Consequently, a script was designed to automate the three Grid-Stat processes into a single action. This script also expanded the automating function to include the WRF forecast data grid remapping process, and replaced the previous manually-intensive and error-prone process. Being more specific, the new script automatically converted the 1.0-km

resolution WRF data to a 2.5-km grid space, converted the 2.5-km resolution RTMA gridded observation to GRIB1 format, set up the directories for the Grid-Stat package, executed the Grid-Stat, and cleaned up the extra files. After the script was successfully tested on the April 23, 2013, 06Z case, all 24 individual hours of the April case study were processed, requiring only about an hour (versus a couple hours) interval of time to process. A copy of these updated scripts is found in appendices D-G.

Executing MODE followed the Grid-Stat assessment. In a second efficiency improvement, the ability to execute multiple MODE runs without exiting the script was developed. With this script, one could more easily process hourly data for multiple variables. Samples of the scripts developed are in appendix D–G.

4.2 MODE Data Preparation – A Boundary Problem

A systematic boundary error was discovered in an early version of the MODE software. As a default, MODE labeled border values as zero then let the statistical algorithms interpret these zero values as “real” data. For the MODE-developer’s principle application to precipitation assessment, this design worked well. However, for continuous variables that cross the zero point (i.e., u-component, v-component), the statistical interpretation produced inaccurate results. MODE experts acknowledged the shortfall, explaining that a code correction would be included with their next MET/MODE release. In the meantime, NCAR suggested changing the default raw-thresh value in the MODE configuration file from “>= 0.0”, to “>= -9999”. The technique was implemented, producing boundary error-free results.

4.3 MODE Data Preparation – Bracketing Data (Thresholding Values) for Finding Objects

The default MODE configuration for describing objects was to define them as a variable $\geq X$, or $= Y$. For precipitation, using the two extremes worked well. With a continuous variable field, however, having the ability to create objects by the bracketing of variable magnitudes in some specified interval within the previous two extremes would provide more useful objects for evaluation. The method developed to do this task had three parts: the first two parts established a lower and upper limit. The lower limit was defined by setting the *conv_thresh* to “>= xxx.x”. The upper limit was set by defining *raw_thresh* as “<= yyy.y”. The third part for bracketing the variable was to skip the convolution step, meaning *conv_radius* was set to zero. The *conv_radius* defines the circular convolution radius applied to smooth the raw fields (Developmental Testbed Center, 2011). The radii are specified in grid units. Because the overall variable range for the domain is so narrow, there is a risk of losing the object if any raw field smoothing were applied. Thus, this action is skipped.

5. Summary and Final Comments

Army decisions regarding the safe and efficient employment of armed forces and equipment are impacted by the atmosphere. Advanced knowledge of atmospheric conditions can empower decision makers to increase mission effectiveness and reduce potential exposure to dangerous scenarios. The ARL has been developing a tool to glean this advanced weather intelligence notification, called, the WRE-N or Nowcast Model. Assessing and improving the “Nowcast” Model’s ability to support battlefield applications is one of ARL’s long term goals. Identifying methods for executing the high-resolution model assessments, and for finding high-resolution observational datasets to evaluate the Nowcast Model, is nontrivial. This report documents an investigation into model verification tools, techniques, and applications relevant to assessing the WRE-N.

Traditional statistical tools are useful for assessing low-resolution forecast output, especially with the aid of largely populated data resources. Some of the standard statistical methods and tools were discussed in the first chapter. Weather conditions, however, rarely repeat themselves exactly; data are often sparse; and weather models being verified must rely on inexact equations when applied on high-resolution grids. Consequently, the traditional assessment methods can be incomplete for characterizing temporal and spatial errors, forcing the assessment process to utilize nontraditional tools.

The MET package developed by NCAR, contains both traditional and nontraditional statistical evaluation tools. ARL’s Model Assessment Project has been conducting an investigation into the applicability of the MET package for ARL’s model V&V mission. In particular, they have been looking into the Method for Object-Based Diagnostic Evaluation tool or “MODE,” as a resource for performing spatial verification of NWP model forecasts. This tool has the potential for quantifying the possible temporal and spatial displacement errors that often occur in high-resolution weather forecasting.

MODE compares meteorological features or “objects” defined from the forecast and observed fields for the same valid time on the basis of measureable attributes. It then quantifies the differences between corresponding objects as a measure of forecast error.

Being more specific, MODE consists of five foundational steps:

1. Define or resolve objects within a forecast and an observation field. Objects are user-defined features within given data fields.
2. Define object attributes, such as the area, centroids, axis angle, and intensity.

3. Compute attribute differences. Examples of attribute differences include an area ratio, centroid distance, angle difference, and intensity ratios between the observation and the forecast objects.
4. Use fuzzy logic to compute the total interest values for each object pair, based on the user-defined weights. Match objects across fields and merge objects within the same field, based on the computed interest values.
5. Summarize the characteristics (output statistics) for the single objects, the object pairs, and the matched/merged objects.

The statistical tool was originally designed for precipitation analyses. ARL's applications, however, require using MODE on continuous variable fields.

Investigating the potential of this tool required first determining if the software installation was valid. This task was assessed by running a "Perfect" case. That is, the observation and forecast data fields were identical. The results of the "Perfect" case confirmed that the software was performing correctly. The task also was a catalyst for developing a technique that can extract data slices from the range of the continuous variable resources. Section 3 provides an overview of this technique and a detailed review of the "perfect case" results.

Running a "real world" case was the next step. The NWP model used was WRE-N (a.k.a. "Nowcast" Model). WRE-N is tailored to address Army-scale horizontal spatial resolutions of 1–3 km. This model was run over three nested grids (9, 3, and 1 km), with the 1-km inner nested grid data as the focus of the "real world" study. Observation data from an independent gridded analysis over the same footprint was weighed against the forecast data.

To assist with the analysis, observational time-series data from a meteorological suite located near the center of the area of interest was reviewed. Significant atmospheric characteristics were flagged for study. While several meteorological variables were reviewed, this report presents a sample of the results; namely, the temperature field at 2-m AGL. The "real world" case forecast time selected was 20 Z, or 1400 LT. Winds were strong, steady and westerly; implying a well mixed environment. The temperature interval that was extracted was 301.00–303.00 K.

MODE results found eight simple objects in the forecast field and two in the observation field. There were 16 matching pairs, with five pairs identified as having a 70% and greater interest level. The simple objects were all merged into one cluster for the MODE analysis. The forecast area grid squares qualifying as a clustered object, were about 25% of the area qualifying in the observation field. Based on user preferences (default settings), the MMI for the forecast was 80.42% and the Observation was 82.73%. The combined MMI (interest) was 80.42%. A visualization of the objects and the larger cluster are in appendix C.

The numerical results between the Perfect and "real world" case were very similar. However, using the visualization, the importance of the area represented in the statistics comes into focus.

Additional case studies need to be examined before the tool can be fully optimized by the model assessment process. Exercising the numerous options in the lengthy configuration file will open up new understandings of both the tool and the V&V results. Examination of the data contained in the other three MODE output files will provide additional results that will contribute to a more complete assessment. With a mastering of the MODE terms and process, this nontraditional tool has the potential to significantly strengthen ARL's NWP model assessment process.

6. References

- Brown, B. G.; Bullock, R. R.; David, C. A.; Gotway, J. H.; Chapman, M. B.; Takacs, A.; Gilleland, E.; Manning, K.; Mahoney, J. 2004: New Verification Approaches for Convective Weather Forecasts. *11th Conf. Aviation, Range, and Aerospace Meteorology*, Hyannis, MA, 4–8 October, 2004.
- Davis, C. A.; Brown, B. G.; Bullock, R.; Halley-Gotway, J. 2009: The Method for Object-Based Diagnostic Evaluation (MODE) Applied to Numerical Forecasts from the NSSL/SPC Spring Program. *Weather and Forecasting*. 2009; **24**, pp. 1252–1267.
- De Pondeva, M. S. F. V.; Manikin, G. S.; Dimego, G.; Benjamin, S. G.; Parrish, D. F.; Purser, R. J.; Wu, W. -S.; Horel, J. D.; Myrick, D. T.; Lin, Y.; Aune, R. M.; Keyser, D.; Colman, B.; Mann, G.; Vavra, J. 2011: The Real-Time Mesoscale Analysis at NOAA’s National Centers for Environmental Prediction: Current Status and Development. *Weather and Forecasting*, 2011; **26**, pp. 593–612.
- Developmental Testbed Center. COPYGBFunctionality.
http://www.dtcenter.org/met/users/support/online_tutorial/METv3.0/index.php (accessed September, 2013).
- Developmental Testbed Center, *Model Evaluation Tools Version 3.0 (METv3.0) User’s Guide 3.0.2*, Boulder, CO, January 2011.
- Ebert, E. E. 2008: Fuzzy Verification of High-Resolution Gridded Forecasts: A Review and Proposed Framework. *Meteorological Applications*. **15**, pp. 51–64.
- Jolliffe, I. T.; Stephenson, D. B. *Forecast Verification: A Practitioner’s Guide in Atmospheric Science*, John Wiley & Sons: Chichester, UK, 2012, pp. 77–117.
- Murphy, A. H. 1993: What is a Good Forecast? An Essay on Nature of Goodness in Weather Forecasting. *Weather Forecasting*; **8**, pp. 281–293.
- National Center for Atmospheric Research (NCAR). *Model Evaluation Tools Version 4.1 User’s Guide*; Developmental Testbed Center: Boulder, CO, May 2013.
- National Weather Service NCEP Central Operations. RTMA Documentation.
http://nomads.ncep.noaa.gov/text_descriptions/RTMA.doc.shtml (accessed September 2013).
- Panofsky, H.; Brier, G. *Some Applications of Statistic to Meteorology*, The Pennsylvania State University, University Park, PA, 1976, pp. 191–208.

Raby, J.; Flanigan, R.; Passner, J.; Dumais, R. *Traditional Statistical Measures Comparing Weather Research and Forecast Model Output to Observations Centered Over Utah*; ARL-TR-5422; U.S. Army Research Laboratory: White Sands Missile Range, NM, 2011.

Raby, J.; Passner, J.; Vaucher, G. *Performance Comparison of High Resolution Weather Research and Forecasting Model Output with North American Mesoscale Model Initialization Grid Forecasts*; ARL-TR-6000; U.S. Army Research Laboratory: White Sands Missile Range, NM, 2012.

Types of Forecasts and Verifications, World Climate Research Programme, WWRP/WGNE Joint Working Group on Forecast Verification Research, 4th International Verification Methods Workshop, Helsinki, Finland, June 4–10, 2009, http://www.cawcr.gov.au/staff/eee/verif/verif_web_page.html#Standard_verification_methods (accessed February 10, 2014).

Wilks, D. *Statistical Methods in the Atmospheric Sciences*; Academic Press: Burlington, MA, 2006.

Appendix A. Example of a MODE Configuration File

The following MODE configuration file example defines numerous parameters needed to run MODE and specify the type and format of the MODE output. As a minimum, user changes employed for testing include the fcst/obs field name (TMP), the fcst/obs field height in meters (Z2), the three parameters needed to extract a slice of the data (conv_thresh, raw-thresh, conv_thresh), and the raw-thresh string, which can be used to distinguish one output from another.

```
////////////////////////////////////
//
// Filename:  WrfModeConfig_m3o3_rtma_WSMR_template
// Last Rev: 130607, jr/gv/yr
//
// Location: carson:MET_MODE
//
// PURPOSE: Sets up how MODE is to be run and specifies the type and
//format of the output.
//
// NOTE:  User inserts the following changes
//      varname      fcst/obs field name.
//      varheight    fcst/obs field height in meters.
//      ctnumLOW     conv_thresh (lowest var threshold)
//      ctnumHI      raw_thresh  (highest var threshold)
//      LOWct        conv_thresh string for output.
//      HIct         raw_thresh string for output.
//      conv_radius  conv_radius from 1 to 3.
//
// MODE configuration file for testing is MET V4.1.
//
// MODE configuration file for testing was MET V4.0.
//
// Generated/modified by JRaby on 11/27/12
// To upgrade from METV3.1 to METV4.0 for use with RTMA gridded
// observations and WRE-N/FDDA, AFWA WRF forecasts in LA basin domains.
//
// Modified on 03/06/2013 (JRaby), to update for use in LA domain 3
//(m3o3)
// which requires the filename change to replace m202 with m3o3 and
// removal of ref to v40test and DPG. Also changes were needed to change
// the paths from METV3.1 to METV4.0 color tables.
//
// For additional information, see the MET_BASE/data/config/README file.
//
////////////////////////////////////
```

This appendix appears in its original form, without editorial change.

```

//
// Output model name to be written
//
model = "WREN";

////////////////////////////////////////////////////////////////

//
// Approximate grid resolution (km)
//
grid_res = 2.5;

////////////////////////////////////////////////////////////////

//
// Forecast and observation fields to be verified
//
fcst = {
  field = {
    name = "varname";
    level = "varheight";
  };

  // raw_thresh           = >=0.0;   - original line, used with the variable
  extreme ranges.
  // raw_thresh           = >=-9999; - gets rid of false boundary data, when
  conv_thresh <=xx.
  // raw_thresh           = <=296.0; - set values > 296.0 to zero (defines
  upper limit).

  // conv_radius           = 1;   - used, when variable extreme ranges (max/min
  only) employed.
  // conv_radius           For example: include all temperatures
  > 290 K only.
  // conv_radius           = 0;   - skips convolution step; used when bracketing
  variable ranges.

  // If find no objects, conv_thresh can be varied to better fit obs and
  fcst data.
  // conv_thresh           = >=5.0;   - original setting, used with variable
  extreme ranges.
  // conv_thresh           = >=295.0; - defines objects >= 295.0K (defines lower
  limit).

  raw_thresh              = ctnumHI;
  conv_radius              = crnum;
  conv_thresh              = ctnumLOW;

  vld_thresh              = 0.5;
  area_thresh              = >=0.0;
  inten_perc_value        = 100;
  inten_perc_thresh       = >=0.0;
  merge_thresh            = >=1.25;
  merge_flag               = THRESH;
};

```

```

obs = fcst;

////////////////////////////////////////////////////////////////

//
// Handle missing data
//
mask_missing_flag = BOTH;

//
// Match objects between the forecast and observation fields
//
match_flag = MERGE_BOTH;

//
// Maximum centroid distance for objects to be compared
//
max_centroid_dist = 125.0/grid_res;

////////////////////////////////////////////////////////////////

//
// Verification masking regions
//
mask = {
    grid      = "";
    grid_flag = NONE; // Apply to NONE, FCST, OBS, or BOTH
    poly      = "";
    poly_flag = NONE; // Apply to NONE, FCST, OBS, or BOTH
};

////////////////////////////////////////////////////////////////

//
// Fuzzy engine weights
//
weight = {
    centroid_dist    = 2.0;
    boundary_dist    = 4.0;
    convex_hull_dist = 0.0;
    angle_diff       = 1.0;
    area_ratio       = 1.0;
    int_area_ratio   = 2.0;
    complexity_ratio = 0.0;
    inten_perc_ratio = 0.0;
    inten_perc_value = 50;
}

////////////////////////////////////////////////////////////////

//
// Fuzzy engine interest functions
//
interest_function = {

    centroid_dist = (

```



```

        (          0.0, 1.0 )
        ( 10.0/grid_res, 1.0 )
        ( 100.0/grid_res, 0.0 )
    );

    boundary_dist = (
        (          0.0, 1.0 )
        ( 60.0/grid_res, 0.0 )
    );

    convex_hull_dist = (
        (          0.0, 1.0 )
        ( 60.0/grid_res, 0.0 )
    );

    angle_diff = (
        ( -0.0, 1.0 )
        ( 30.0, 1.0 )
        ( 90.0, 0.0 )
    );

    corner      = 0.8;
    ratio_if = (
        ( -0.0, 0.0 )
        ( corner, 1.0 )
        ( 1.0, 1.0 )
    );

    area_ratio = ratio_if;

    int_area_ratio = (
        ( 0.00, 0.00 )
        ( 0.10, 0.50 )
        ( 0.25, 1.00 )
        ( 1.00, 1.00 )
    );

    complexity_ratio = ratio_if;

    inten_perc_ratio = ratio_if;
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//
// Total interest threshold for determining matches
//
total_interest_thresh = 0.7;

//
// Interest threshold for printing output pair information
//
print_interest_thresh = 0.0;

////////////////////////////////////////////////////////////////////////////////////////////////////////////////

```

```

//
// Plotting information
//
met_data_dir = "/home/rflaniga/opt/MET/MET_4.0_patch/data";

fcst_raw_plot = {
    color_table      =
"/home/rflaniga/opt/MET/MET_4.0_patch/data/colortables/met_default.ctable"
;
    plot_min         = 0.0;
    plot_max         = 0.0;
    colorbar_spacing = 1;
};

obs_raw_plot = {
    color_table      =
"/home/rflaniga/opt/MET/MET_4.0_patch/data/colortables/met_default.ctable"
;
    plot_min         = 0.0;
    plot_max         = 0.0;
    colorbar_spacing = 1;
};

object_plot = {
    color_table      =
"/home/rflaniga/opt/MET/MET_4.0_patch/data/colortables/mode_obj.ctable";
};

//
// Number of grid boxes to fill with bad data values along the edge of the
// field
// to avoid edge effects.
//
zero_border_size = 1;

//
// Boolean for plotting on the region of valid data within the domain
//
plot_valid_flag = FALSE;

//
// Plot polyline edges using great circle arcs instead of straight lines
//
plot_gcArc_flag = FALSE;

////////////////////////////////////

//
// NetCDF matched pairs, PostScript, and contingency table output files
//
ps_plot_flag      = TRUE;
nc_pairs_flag     = TRUE;
ct_stats_flag     = TRUE;

////////////////////////////////////

```

```
output_prefix = "m3o3WRENwsmrvarnamehtstringm_LOWct_HIct_";  
// version      = "V4.0"; Old version
```

```
version      = "V4.1";
```

```
////////////////////////////////////
```

Appendix B. “Perfect” Case Post-Script File Output

The “Perfect” Case uses the same observation data file, as the “forecast” data file. The following example of a “Perfect” Case used observation data from April 23, 2013. These data were initiated and valid at 2000 UTC (1400 LT). The output consists of seven figures—B-1 through B-7.

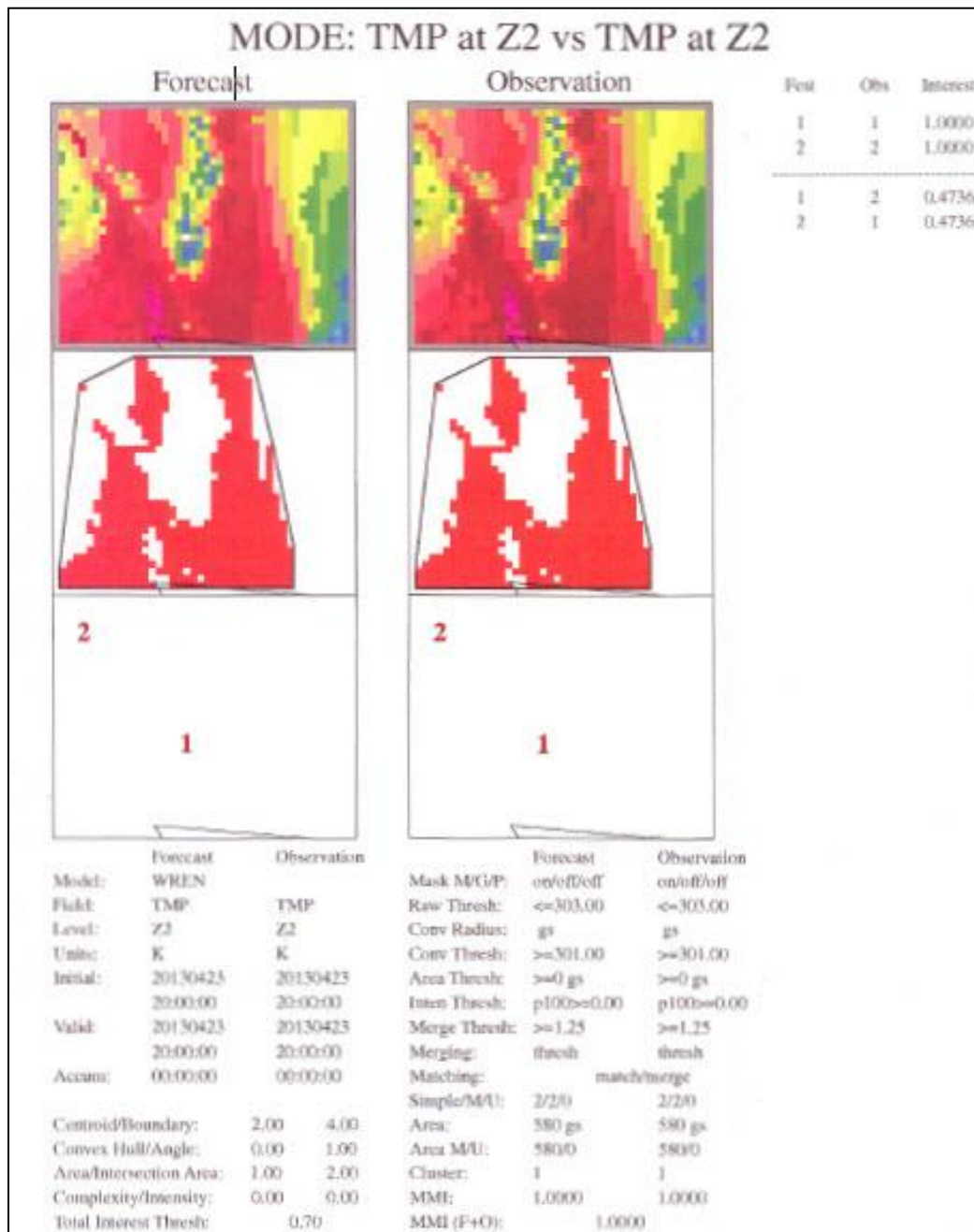


Figure B-1. “Perfect” Case Summary page, as described in section 3.3.3.

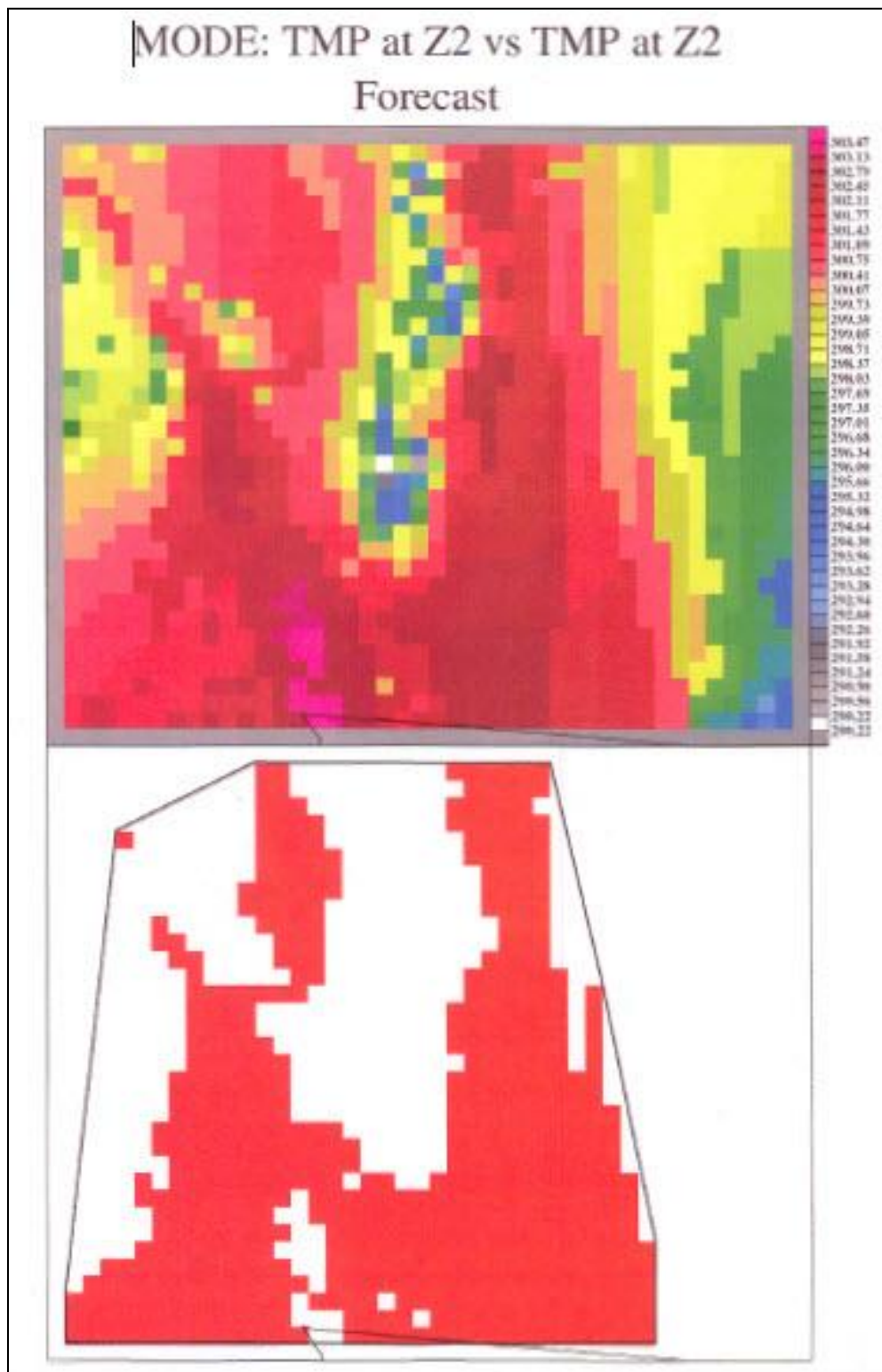


Figure B-2. The “Perfect” Case Forecast raw data are graphically displayed on the top, with a color key legend on the right. The graphical display of the “Perfect” Case forecasted merged and matched objects is on the bottom of the page.

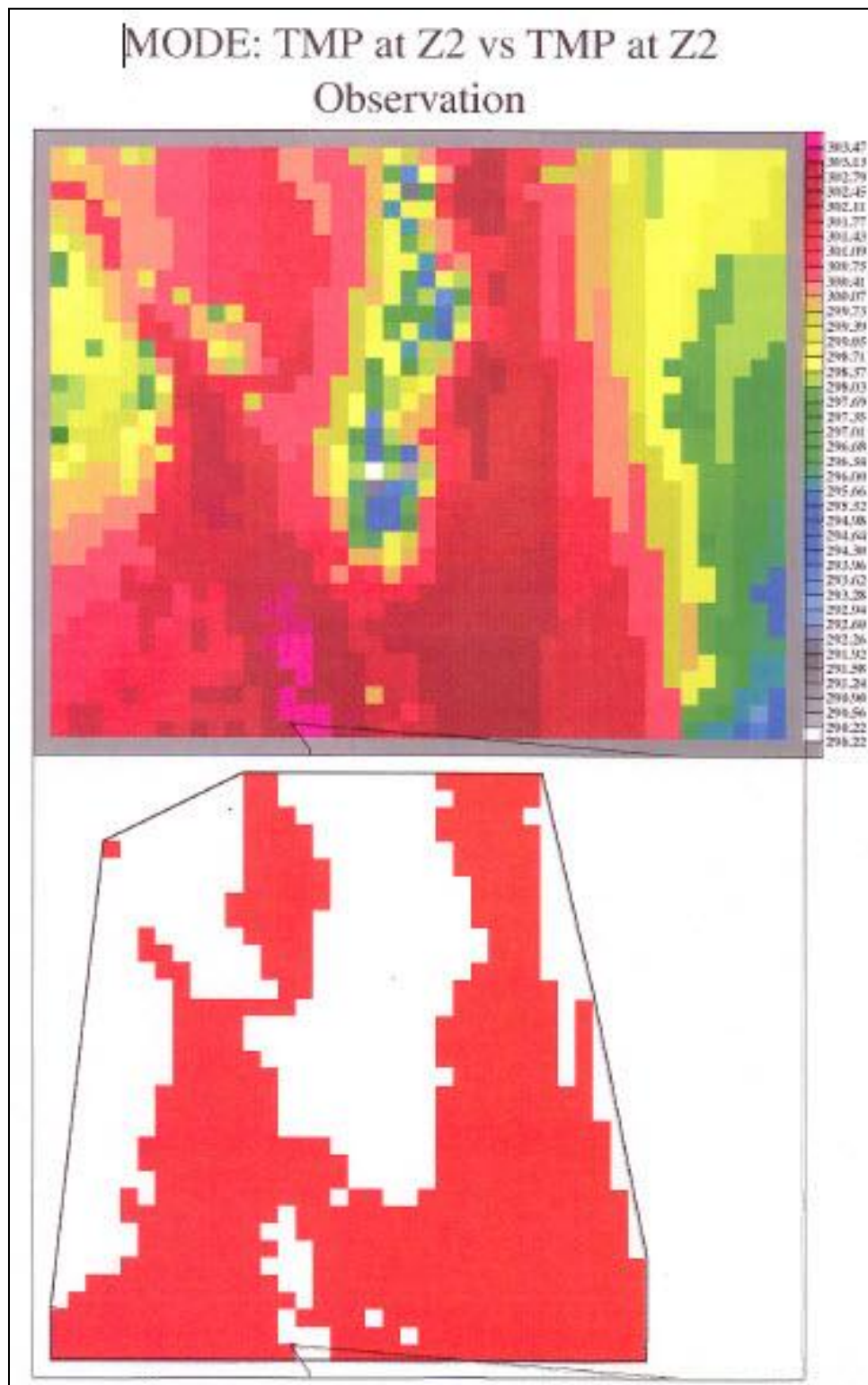


Figure B-3. The “Perfect” Case Observation data are graphically displayed on the top, with a color key legend on the right. The graphical display of the observed merged and matched objects is on the bottom of the page. Since a “perfect case” was used, there is no difference between figures B-2 and B-3.

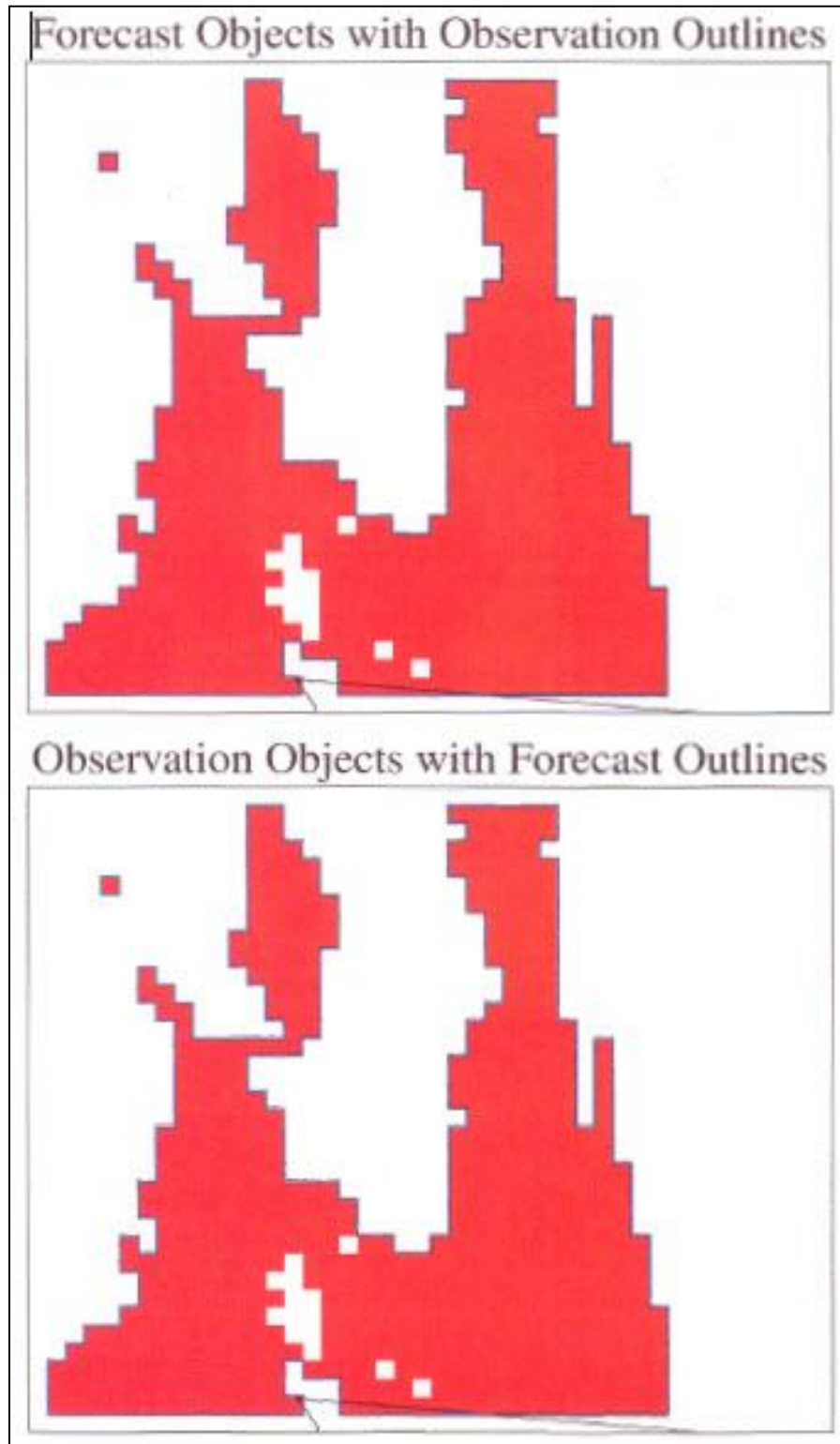


Figure B-4. Top-diagram provides a graphical display of the “Perfect” Case forecast objects in color and the observation objects outlined in black. The bottom diagram reverses the pattern, placing the “Perfect” Case observation objects in color and “Perfect” Case forecast objects in a black outline.

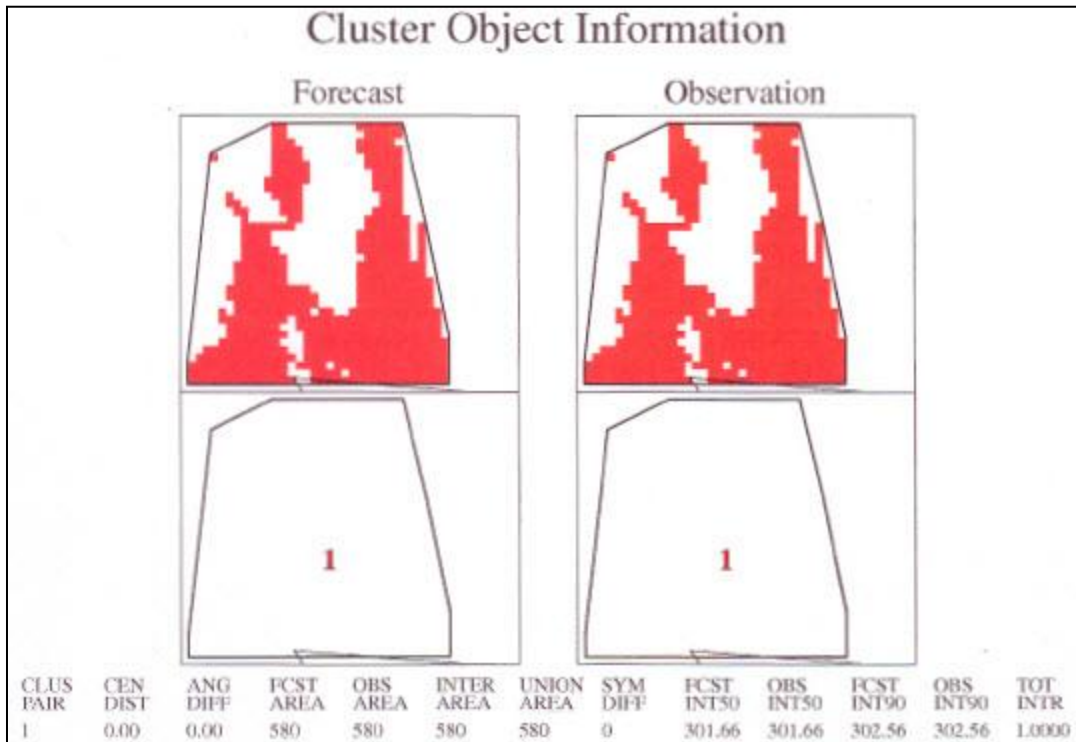


Figure B-5. The “Perfect” Case cluster object information is displayed and tabulated. Since the case was perfect, the *centroid distance* and *angle differences* are zero grid units and degrees, respectively. The forecast and observation areas (in pixels), intersection area (in grid squares) and union area (in grid squares) are also identical magnitudes. Consequently, the symmetric differences of the objects are zero. The FCST and OBS intensity percentile variable corresponds to the intensity ratio weight variable. Total interest is 1.0000 for all clusters.

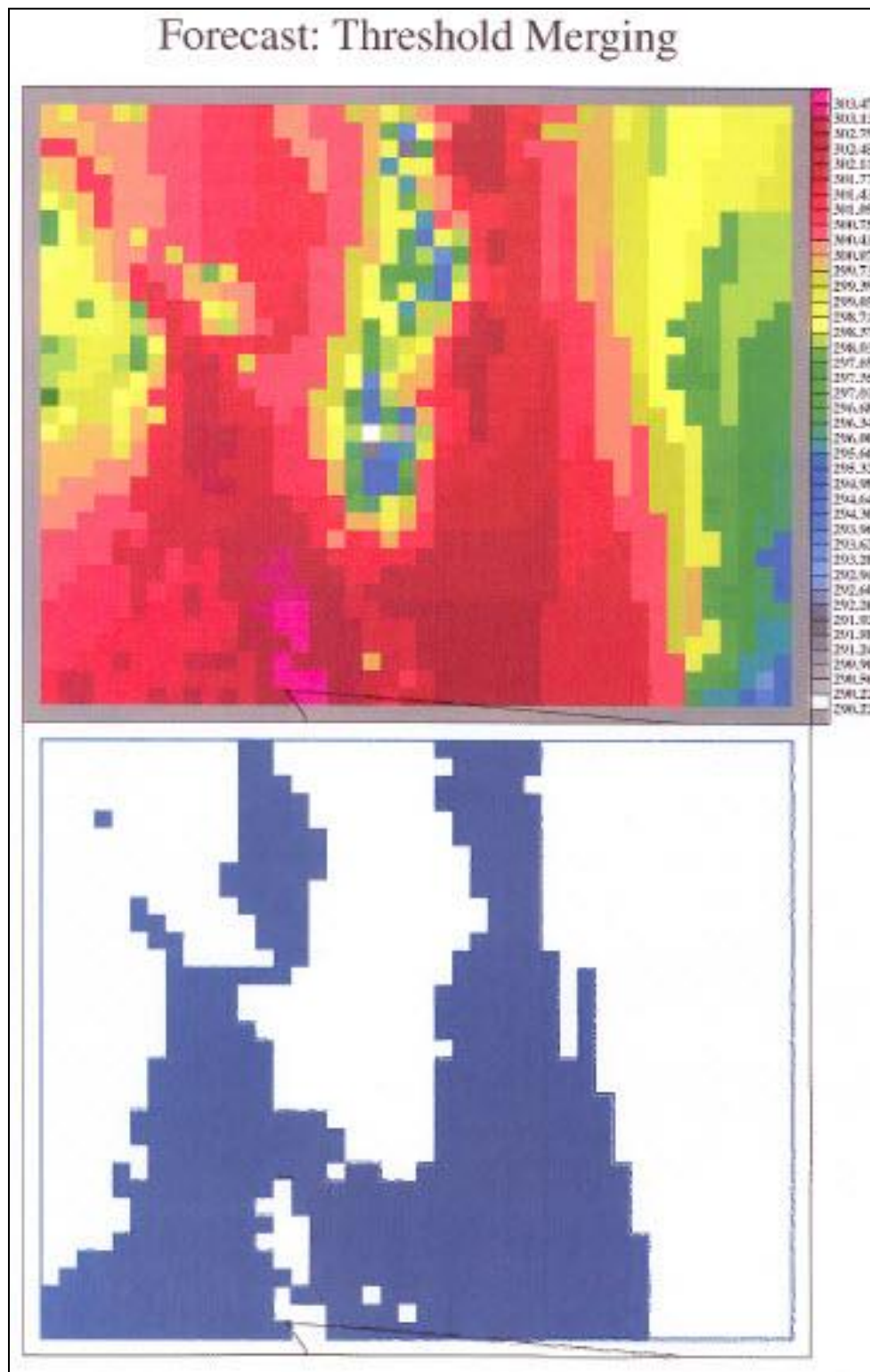


Figure B-6. The “Perfect” Case forecast *Threshold Merging* data are shown.

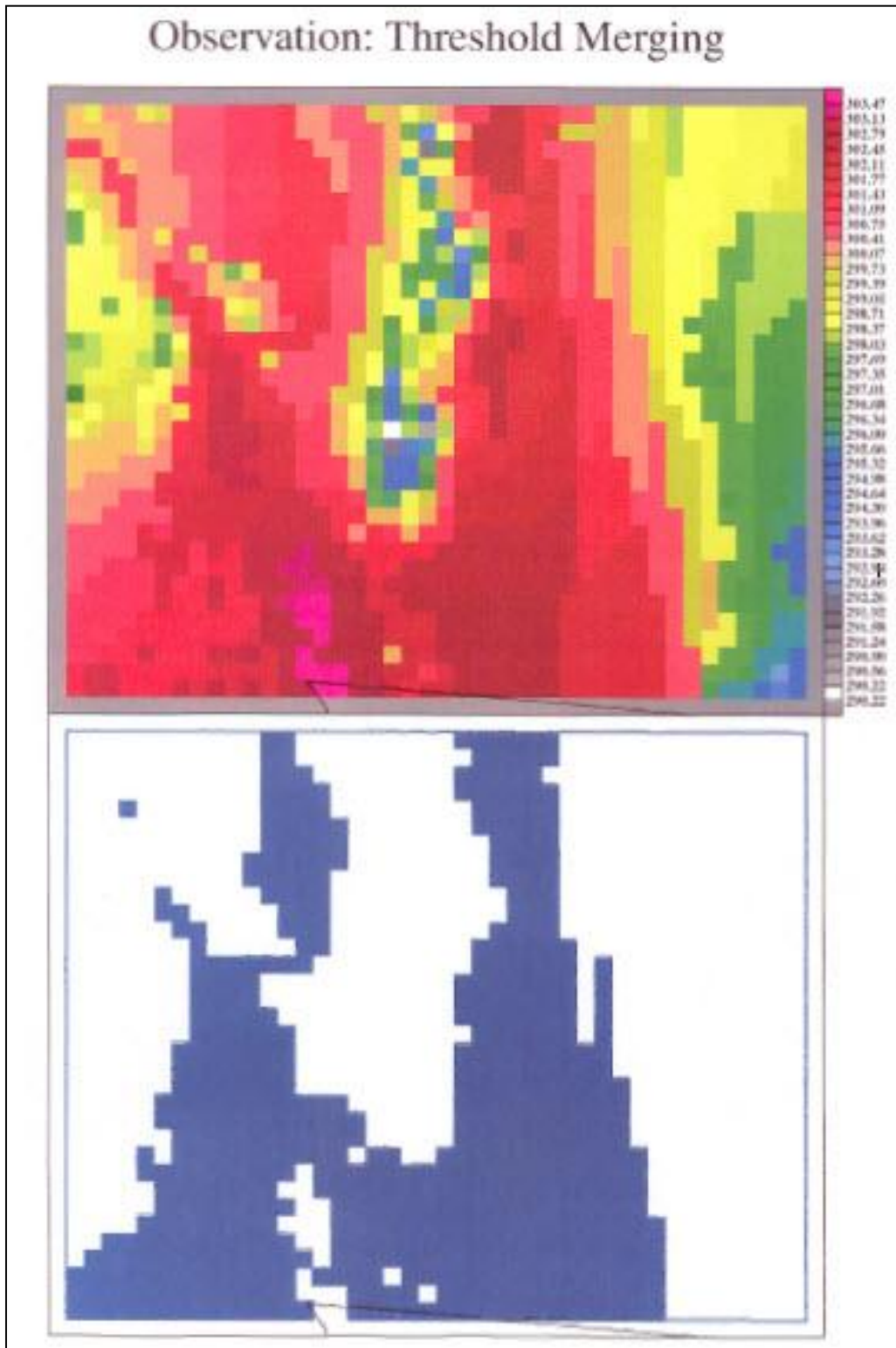


Figure B-7. The “perfect case” Observation *Threshold Merging* data are shown. The “perfect case” scenario generates identical merged images (bottom plot) for figures B-6 and B-7.

INTENTIONALLY LEFT BLANK.

Appendix C. “Real World” Case Post-Script File Output

The “real world” case uses forecast and observation data resources for April 23, 2013. These data were initiated at 0600 UTC and valid at 2000 UTC (1400 LT). The output consists of seven figures—C-1 through C-7.

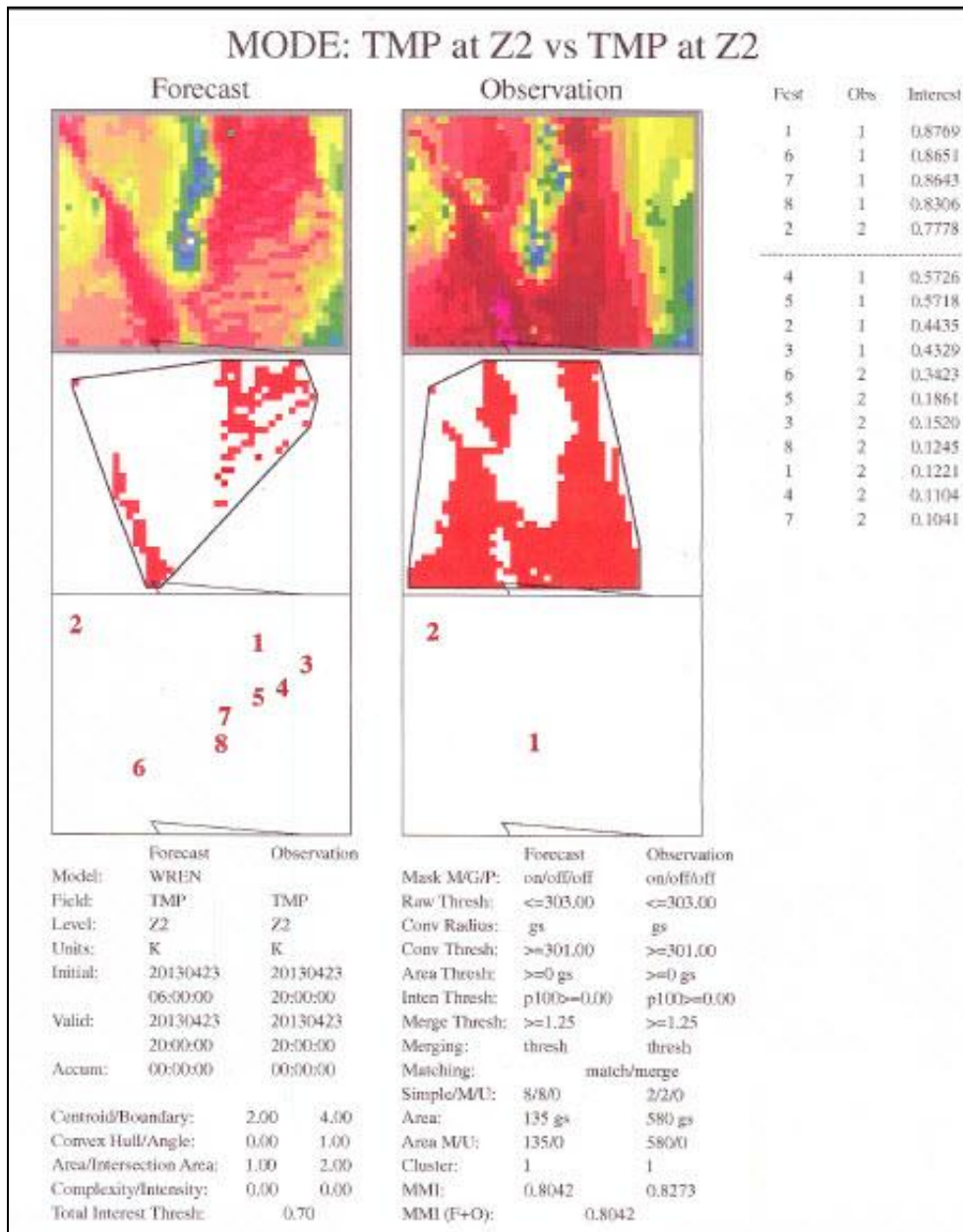


Figure C-1. The “real world” case summary page, as described in section 3.4.

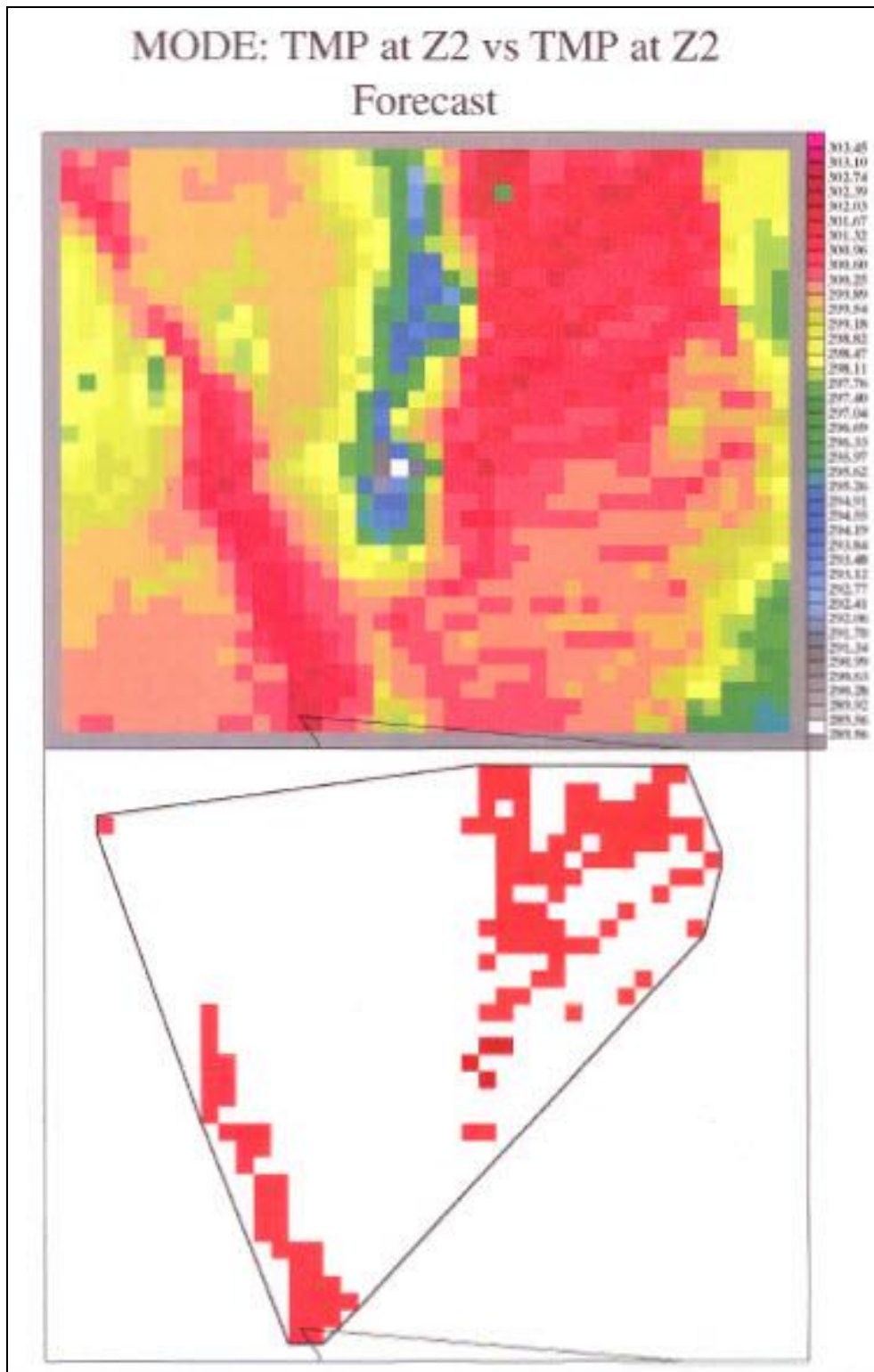


Figure C-2. The “real world” case roforecast raw data are graphically displayed on the top, with a color key legend on the right. The graphical display of the forecasted merged and matched objects is on the bottom of the page.

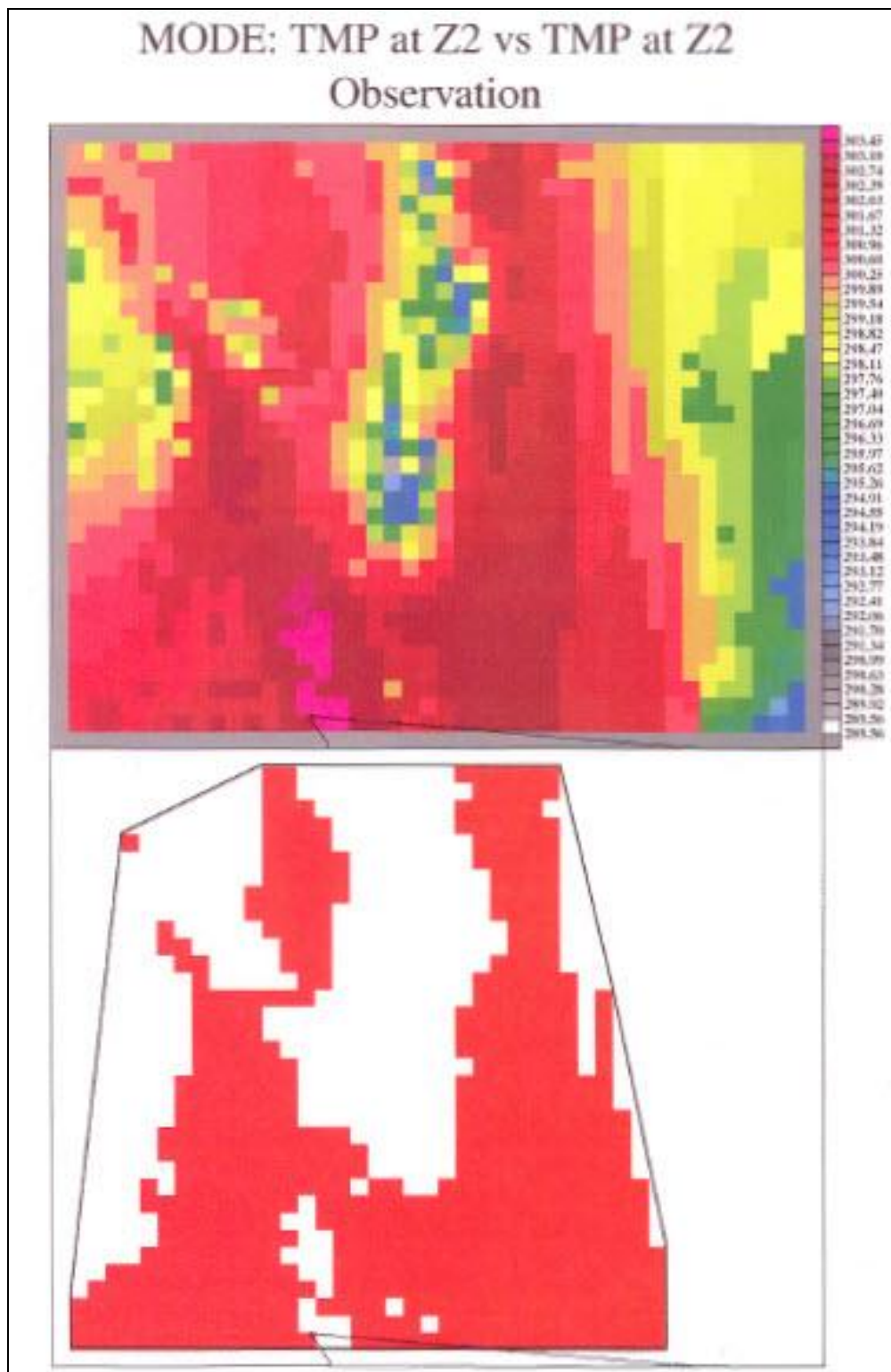


Figure C-3. The “real world” case observation data are graphically displayed on the top, with a color key legend on the right. The graphical display of the observed merged and matched objects is on the bottom of the page.

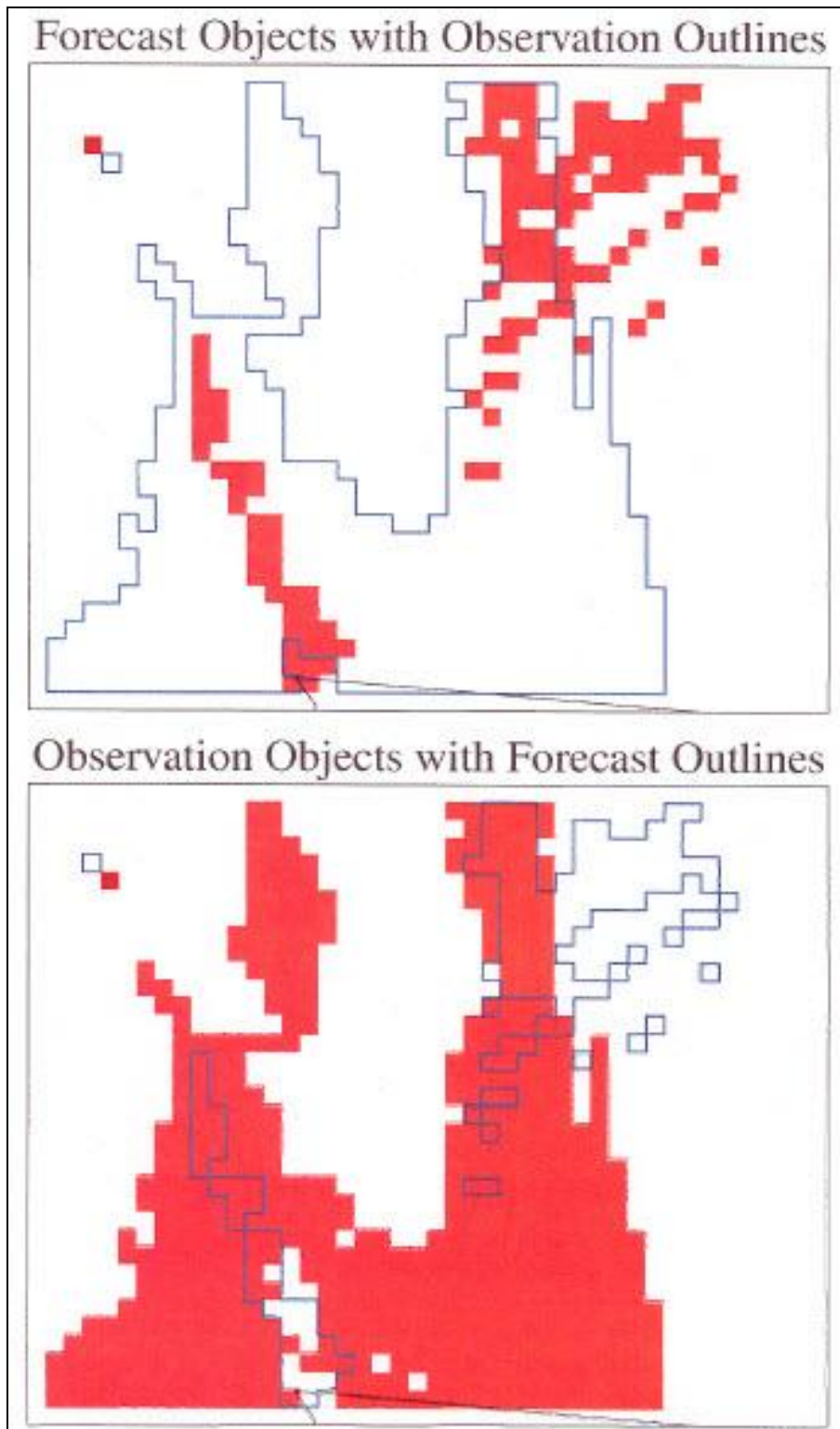


Figure C-4. Top-diagram provides a graphical display of the “real world” case forecast objects in color and the observation objects outlined in black. The bottom diagram reverses the pattern, placing the observation objects in color and forecast objects in a black outline.

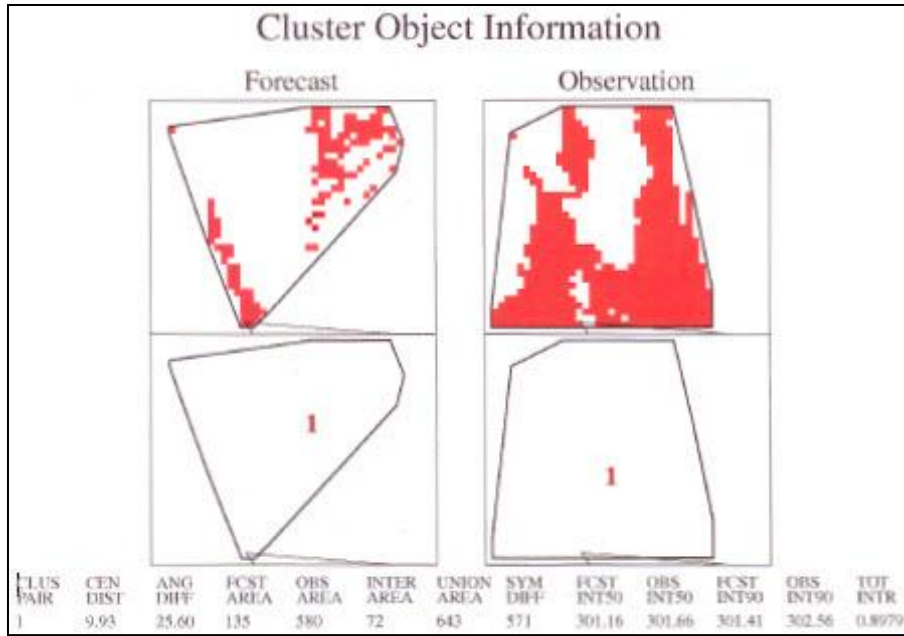


Figure C-5. The “real world” case cluster object information is displayed and tabulated.

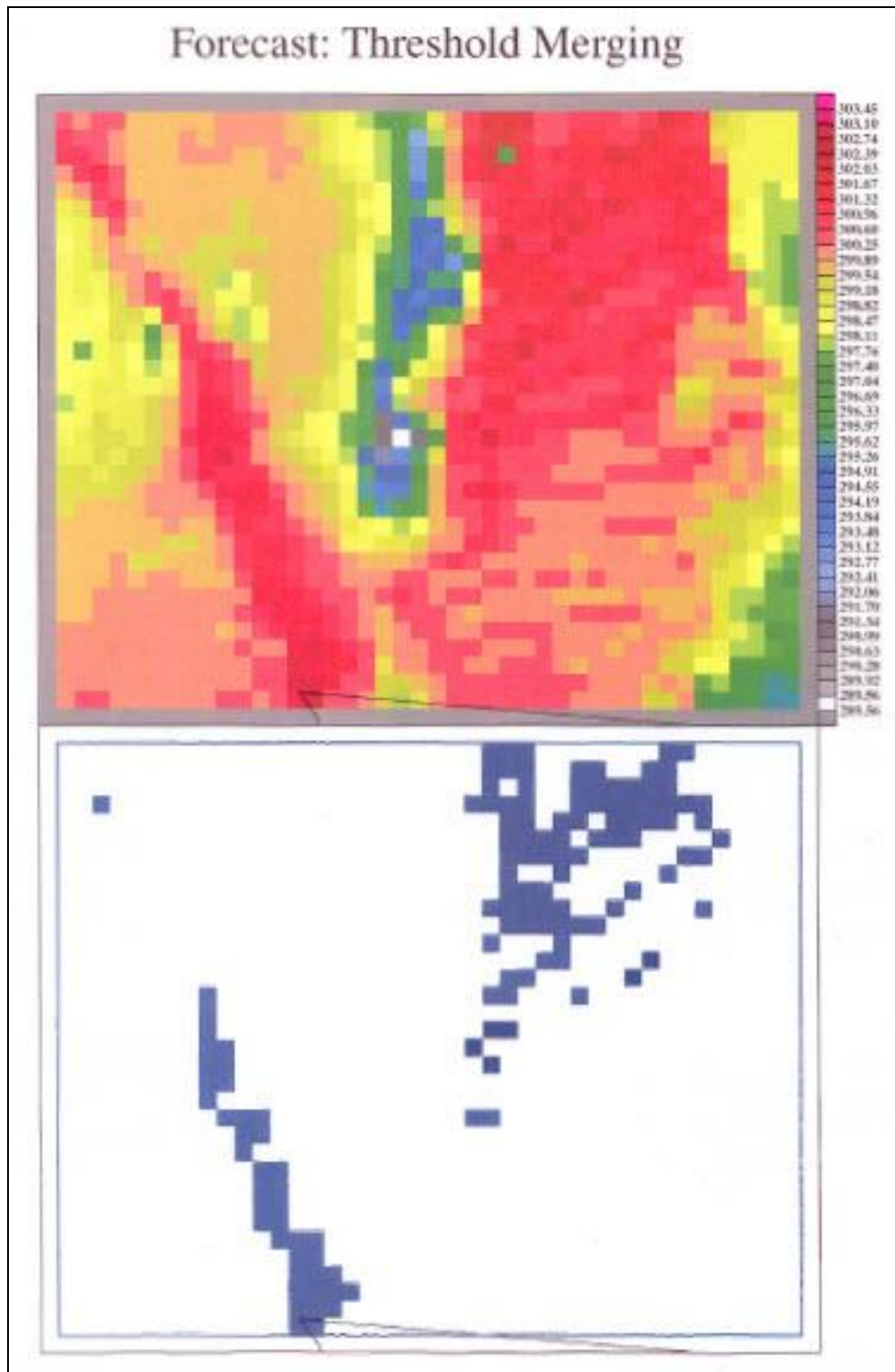


Figure C-6. The “real world” case forecast *Threshold Merging* data are shown.

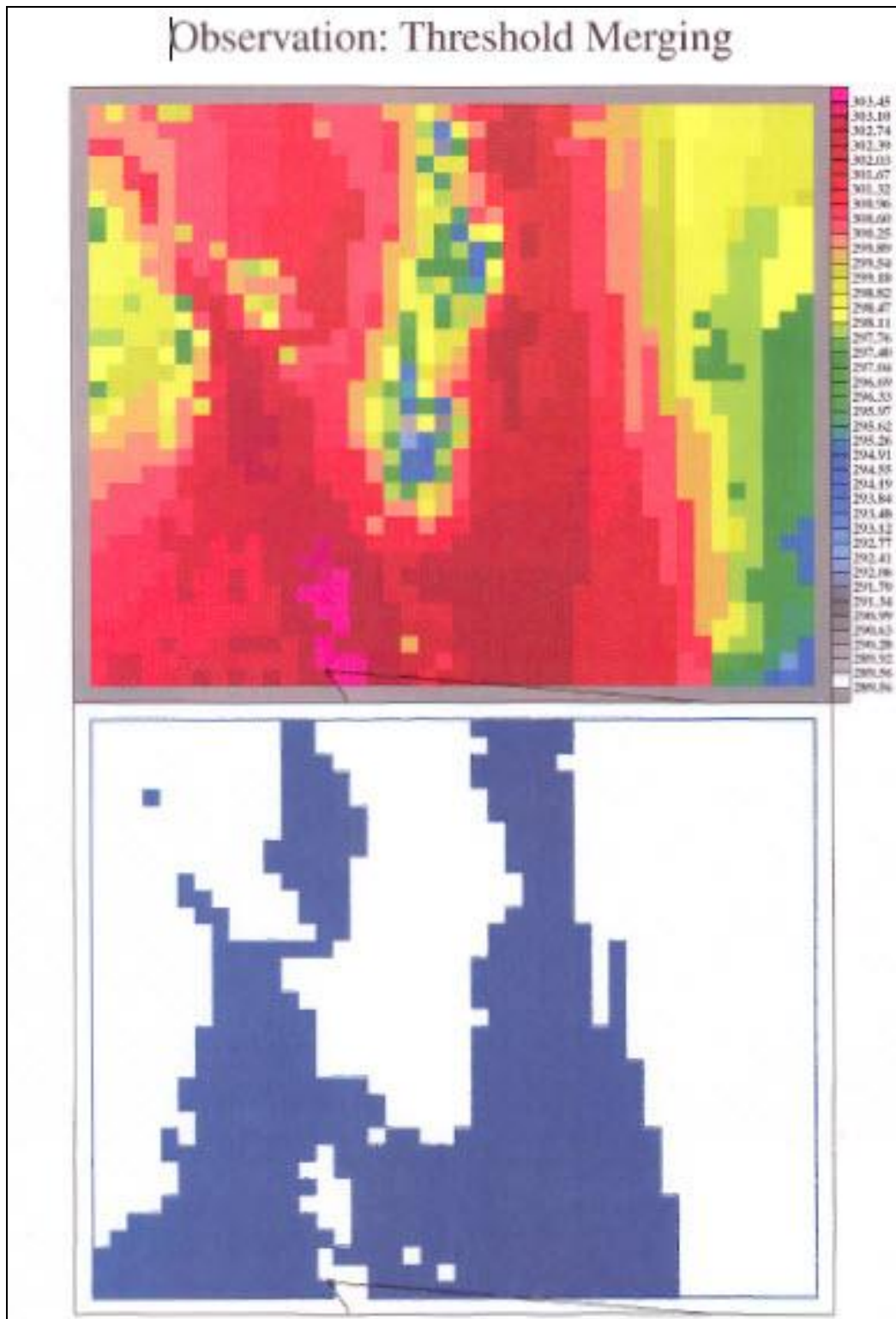


Figure C-7. The “real world” case observation *Threshold Merging* data are shown.

INTENTIONALLY LEFT BLANK.

Appendix D. MET and MODE Script *s_multi*

s_multi is a top level start script that runs WRF Initiation, Point-Stat, Grid-Stat and MODE. The script is menu-driven and takes the user selection as a keyboard entry. The code is designed to facilitate multiple runs of the top-level start script (such as for a 24 1-h periods). The script calls other scripts, which are listed in the commentary.

```
# Filename:  s_multi
# Last Rev:  130517, gv/jr
#
# Location:  carson
#
# PURPOSE:  Top Level start script which
#           Runs scripts for various assessment processes including WRF
#           Initialization, Point-Stat, Grid-Stat, MODE.
#           Does multiple runs of s (as in 24 hr cycle can be run).
#
# Author:   Brown/Raby
# Script Calls:
#   WRF_Main, Create_Passner_Directories,
#   run_prepBUFR, run_MADIS, run_MADIS_Archive,
#   ascii2netcdf,
#   run_Point_Stat,run_Stat_Analysis,run_ExtractStatAnalysis,
#   Set_up_GridStat, run_Grid_Stat, File_cleanup,
#   run_MODE,
#   copy_harold, copy_carson.
#

clear

echo "
"
echo "          MODEL ASSESSMENT"
echo "
"
echo "Enter number of task"
echo "  1 Run WRF Initialization"
echo "  2 Create Passner Directories"
echo "  3 Copy WRF Initialization files from carson to harold"
echo "  4 Copy Post Processed WRF Ouput Files from harold to carson"
echo "  5 Convert prepBUFR Data to netcdf format (metar, synoptic and
upper air)"
echo " "
echo "  6 Download MADIS Current Data (mesonet data)"
echo "  7 Download MADIS Archived Data (archived mesonet data)"
echo "  8 Convert MADIS ASCII data to netcdf"
echo " "
```

This appendix appears in its original form, without editorial change.

```

echo " 9 Run Point-Stat"
echo " 10 Run Stat-Analysis"
echo " 11 Run Stat-Analysis data extraction"
echo " 12 GridStat: Set up data, run Grid-Stat and clean up files"
echo " 13 Disabled - Run Grid-Stat"
echo " 14 Disabled - Cleanup files after Grid-Stat"
echo " 15 Run MODE"
echo " 16 Edit Scripts"
echo " 17 Quit"
echo " "
echo " "
echo " "
echo " "
echo " "
read response
case $response in # Start of case
(1)
echo " "
echo " "
echo "Running WRF_Main"
echo " "
echo " "
WRF_Main
;;

(2)
echo " "
echo " "
echo "Creating Directories for Passner WRF runs for all Settings on
Carson"
echo " "
echo " "
Create_Passner_Directories
;;

(3)
echo " "
echo " "
echo "Copy WRF initialization files from carson to harold"
echo " "
echo " "
copy_harold
;;

(4)
echo " "
echo " "
echo "Copy post-processed WRF output files from harold to carson"
echo " "
echo " "
copy_carson
;;

(5)
echo " "

```

```

echo " "
echo "Converting prepBUFR Data containing metar, synoptic and upper air"
echo " "
echo " "
run_prepBUFR
;;

(6)
echo " "
echo " "
echo "Downloading MADIS Current Data containing mesonet data"
echo " "
echo " "
run_MADIS
;;

(7)
echo " "
echo " "
echo "Downloading MADIS Archived Data containing mesonet data"
echo " "
echo " "
run_MADIS_Archive
;;

(8)
echo " "
echo " "
echo "Converting MADIS ASCII data to netcdf"
echo " "
echo " "
ascii2netcdf
;;

(9)
echo " "
echo " "
echo "Running Point-Stat"
echo " "
echo " "
run_Point_Stat
;;

(10)
echo " "
echo " "
echo "Running Stat-Analysis"
echo " "
echo " "
run_Stat_Analysis
;;

(11)
echo " "

```

```

echo " "
echo "Running Stat-Analysis data extraction"
echo " "
echo " "
run_ExtractStatAnalysis
;;

(12)
echo " "
echo " "
echo "GridStat_multi: Set up data, run Grid-Stat and clean up files"
echo " "
echo " "

clear

# GS-STEP 0: Required input - Start Date and Time(s).
# Enter Start Date of WRF run
echo " "
echo " "
echo "Enter Start_Date (YYYYmdd) of the completed WRF run"
read Start_Date

echo $Start_Date
echo " "

echo "Enter 2-digit, zulu, observation hour (HH) for remapping"
read oHH

echo "Enter 2-digit, forecast validation hour (hh) for remapping"
read fhh

# GS-STEP 1: Convert WRF data to 2.5 km, and setup directories/files
cd ~/Scripts/GridStat_1Script
Set_up_GridStat_multi $Start_Date $oHH $fhh

# GS-STEP 2: Run Grid Stat
cd ~/Scripts/GridStat_1Script
run_Grid_Stat_multi $Start_Date

# GS-STEP 3: Clean up Grid Stat processing files
~/Scripts/GridStat_1Script/File_cleanup_multi $Start_Date $oHH
echo " "
;;

(13)
echo " "
echo " "
echo "Running Grid-Stat - disabled"
echo " "
echo " "
run_Grid_Stat
;;

```

```
(14)
echo " "
echo " "
echo "Running File cleanup after Grid-Stat-disabled"
echo " "
echo " "
File_cleanup
;;

(15)
echo " "
echo " "
echo "Running MODE"
echo " "
echo " "
run_MODE_2
;;

(16)
cd Scripts
clear
ls
echo "Enter the name of script to edit"
read response2
vi $response2
;;

(17) exit 0
;;

esac # end of case -----
```


INTENTIONALLY LEFT BLANK.

Appendix E. MET and MODE Script *run_MODE_2*

run_MODE_2 script hardwires two functions: one for the standard analysis case (forecast versus observations) and one for a “Perfect” Case (observations versus observations). When executed, the user selects either the standard analysis case or the “Perfect” Case function. The script is embedded in and called by *s_multi*.

```
# Filename: run_MODE_2
# Last Rev: 130618, jr/yr/gv
#
# Location: carson:/Scripts
#
# PURPOSE: Control script which sets up the MET MODE tool to ingest the
#          post processed WRF output and RTMA gridded observations.
#          Two options: Standard(fcst vs obs)& Perfect cases (obs vs obs)
#          Options are hardwired.
#-----
cd ~gvaucher/MET_MODE
echo "Enter the Start Date of the post processed WRF data"
read Start_Date
echo " "
echo "Start_Date Entered:" $Start_Date
echo " "
echo "Enter the WRF/RTMA forecast hour, followed by letter Z [##Z]"
read Hour
echo " "
echo "Forecast hour Entered:" $Hour
echo " "

#echo "Running MODE for m1o1"
#run_MODE_m1o1.sh ${Start_Date}

# For Standard Cases (fcst vs obs) use this script:
echo "Running MODE for m3o3, Standard Cases..."
echo " "
run_MODE_m3o3_2.sh ${Start_Date} ${Hour}

# For Perfect Cases (obs vs obs) use this script
# echo "Running MODE for m3o3, Perfect Case (obs vs obs)..."
# echo " "
# run_MODE_m3o3_2_Perfect.sh ${Start_Date} ${Hour}

echo " "
echo "run_MODE Complete."
echo " "
```

This appendix appears in its original form, without editorial change.

INTENTIONALLY LEFT BLANK.

Appendix F. MET and MODE Script *run_MODE_m303_2.sh*

run_MODE_m303_2.sh runs MET_MODE on post-processed 1-km resolution WRF forecast data that have been remapped to a 2.5-km grid over domain 3, using 2.5-km RTMA product as the gridded observations dataset. Note: The script's initiating command requires a start_date and hour. The script is embedded in and called by *run_MODE_2*.

```
# Filename: run_MODE_m3o3_2.sh
# Last Rev: 130610, gv/jr/yr
#
# Location: carson:/MET_MODE
#
# PURPOSE: Sets up and runs MET MODE on post-processed 1km res WRF
# forecast data which has been remapped to a 2.5 km grid over WSMR
# domain 3 and a 2.5 km RTMA product as the gridded observations
# dataset.
#
# Support files needed to execute this script:
#   WrfModeConfig_m3o3_rtma_WSMR_template
#
# Output includes a configuration file with the variable acronym at the
# end:
#   WrfModeConfig_m3o3_rtma_WSMR_TMP
#   WrfModeConfig_m3o3_rtma_WSMR_DPT
#   WrfModeConfig_m3o3_rtma_WSMR_WIND
#   WrfModeConfig_m3o3_rtma_WSMR_UGRD
#
# MODE results are in: /home/gvaucher/MET_MODE/results_m3o3
#
# Script called by: s_multi (select #15-Run MODE)
#
# Script OUTLINE:
#   1. User Input: Select variable.
#   2. Define variable name and variable height.
#       Script converts variable thresholds to strings for output file
name.
#   3a. User enter object boundary definition.
#   3b. Script and User Define object boundaries.
#   4. Convert MODE configuration template into user-defined config file.
#   5. Run MODE using the user-defined specifications.
#
# =====
#
Start_Date=$1
Hour=$2
mkdir -p ./results_m3o3/${Start_Date}
clear
```

This appendix appears in its original form, without editorial change.

```

echo "Running MODE..."
echo "  "
echo "  "
echo "  "

# *****
# 1. User Input: Select variable.
# *****

# This runs MODE on 2.5 km, WRF forecast & 2.5 km RTMA obs for testing.

echo "Enter number of task"
echo "  1 Temperature           (2m AGL)"
echo "  2 Dewpoint Temperature (2m AGL)"
echo "  3 Wind                    (10m AGL)"
echo "  4 U-component             (10m AGL)"
echo "  9 Exit"
echo "  "
echo "  "

read response1

# *****
# 2. Define variable name and variable height.
# *****

case $response1 in # Start of case
(1)
# Code for TMP.
varname="TMP"
varheight="Z2"
htstring="2"

;;

(2)
# Code for Dewpoint - DPT
varname="DPT"
varheight="Z2"
htstring="2"
;;

(3)
# Code for WIND
varname="WIND"
varheight="Z10"
htstring="10"
;;

(4)
# Code for UGRD (u-component)
varname="UGRD"
varheight="Z10"
htstring="10"

```

```

;;

(9)
# Exit the Code
echo " "
echo "Exiting MODE."
echo " "
exit 0
;;

esac #end of case

# *****
# 3a. User enter object boundary definition.
# *****

echo " "
echo "Define object boundaries:"
echo " a. Define UPPER and LOWER variable boundaries."
echo " b. Define boundary as all magnitudes GREATER THAN or EQUAL TO XX"
echo " c. Define boundary as all magnitudes LESS THAN or EQUAL TO XX"
echo " z. Exit"
echo " "

read response2

echo " "
# *****
# 3b. Script and User Define object boundaries.
# *****

case $response2 in # Start of case
(a)
# Define UPPER and LOWER variable boundaries.

# a.1 raw_thresh - Define UPPER limit value.
# ctnumHI is "raw_thresh" in template.
# Sets values > raw_thresh to zero.
#
echo "Enter object's HIGHEST boundary variable value (xxx.x) in kelvin or
m/s."
echo " "
echo " This raw_thresh entry will be converted to: <=xxx.x "

read ctnumHI
echo " "

# Converts Highest Value to a string reference for the output file.
HIct=${ctnumHI/./p}
HIct="le"${HIct}

ctnumHI="<="${ctnumHI}

# a.2 conv_thresh - Define LOWER limit value.

```

```

#   ctnumLOW is "conv_thresh" in template.
#   Defines objects >= xxx.x
#   conv_thresh is set by user.
#

echo " "
echo "Enter object's LOWEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: >=xxx.x "

read ctnumLOW
echo " "

# Converts Lowest Value to a string reference for the output file.
LOWct=${ctnumLOW/./p}
LOWct="ge"${LOWct}

ctnumLOW=">=${ctnumLOW}

# a.3 conv_radius
#   crnum is "conv_radius" in template.
#   Define conv_radius as 0 (skips convolution step, since
#       raw_thresh introduced zeros).
#       For max/min extremes, conv-radius is 1 grid unit.

crnum=0

# if [ $ctnumLOW = $ctnumHI ]
#   then
#     crnum=1
# fi
;;

(b)
# Define boundary as all magnitudes GREATER THAN or EQUAL TO XX.

# b.1 raw_thresh - Define UPPER limit value.
#   ctnumHI is "raw_thresh" in template.
#   raw-thresh is set to >=-9999; to avoid false boundary data.
#   UPPER value is infinity.

ctnumHI=">=-9999"
HIct="9999"

# b.2 conv_thresh - Define LOWER limit Value.
#   ctnumLOW is "conv_thresh" in template.
#   Defines objects >= xxx.x
#   conv_thresh is set by user.

# Define LOWER value.
echo " "
echo "Enter object's LOWEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: >=xxx.x "

```

```

read ctnumLOW
echo " "

# Converts Lowest Value to a string reference for the output file.
LOWct=${ctnumLOW/./p}
LOWct="ge"${LOWct}

ctnumLOW=">=${ctnumLOW}

# b.3 conv_radius
#   crnum is "conv_radius" in template.
#   conv_radius is set to 0, which skips the smoothing step.
#   conv_radius set to 1 smooths data by 1 grid unit.
crnum=0
;;

(c)
# Define boundary as all magnitudes LESS THAN or EQUAL TO XX.

# c.1 raw_thresh - Define LOWER limit value.
#   ctnumHI is "raw_thresh" in template.
#   raw-thresh is set to >=-9999; to avoid false boundary data.
#   LOWEST value is infinity.
#
ctnumHI=">=-9999"
HIct="9999"

# c.2 conv_thresh - Define HIGHEST limit value.
#   ctnumLOW is "conv_thresh" in template.
#   Defines objects <= xxx.x
#   conv_thresh is set by user.
#
# Define HIGHEST value.
echo " "
echo "Enter object's HIGHEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: <=xxx.x "

read ctnumLOW
echo " "

# Converts this highest Value to a string reference for output file.
LOWct=${ctnumLOW/./p}
LOWct="le"${LOWct}

ctnumLOW="<=${ctnumLOW}

# c.3 conv_radius
#   crnum is "conv_radius" in template.
#   conv_radius is set to 0, which skips the smoothing step.
#   conv_radius set to 1 smooths data by 1 grid unit.

crnum=0

```



```

;;

(z)
# Exit the Code
echo " "
echo "Exiting MODE."
echo " "
exit 0
;;

esac #end of case

echo "raw_thresh [aka ctnumHI] = "$ctnumHI" "
echo "conv_radius [aka crnum] = "$crnum" "
echo "conv_thresh [aka ctnumLOW] = "$ctnumLOW" "
echo " "
echo "LOWct = "$LOWct" "
echo "HIct = "$HIct" "
echo " "

# *****
# 4. Convert MODE configuration template into user-defined config file.
# (Old file is deleted.)
# *****
#

echo "Removing old WrfModeConfig_m3o3_rtma_WSMR_${varname} file. Is this
ok?"
read YN

if [ $YN = "n" ]
then
echo "Exiting script. (At least in theory...)"
exit 0
fi

rm ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

cp ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_template
~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/${varname}/${varname}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>
~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/${varheight}/${varheight}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varna
me}1>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/${ctnumLOW}/${ctnumLOW}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname
}>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/${ctnumHI}/${ctnumHI}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

```

```

sed
s/LOWct/${LOWct}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/HIct/${HIct}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/crnum/${crnum}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/htstring/${htstring}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

rm ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1

ls -al WrfModeConfig_m3o3_rtma_WSMR_${varname}*

# exit

# *****
# 5. Run MODE using the user-defined specifications.
# *****
#
mode ../MET_WRFpostprd/${Start_Date}/wrf_2p5km_${Hour}.grb1
../MET_obs/RTMA/${Start_Date}/rtma_2p5km_${Hour}.grb1
../WrfModeConfig_m3o3_rtma_WSMR_${varname} -outdir
./results_m3o3/${Start_Date} -log ./logs/test -v 2

# Run completed!
echo " "
echo "<run_MODE_m3o3_2.sh> Complete."
echo " "

exit

```

INTENTIONALLY LEFT BLANK.

Appendix G. MET and MODE Script *run_MODE_m303_2_Perfect.sh*

run_MODE_m303_2_Perfect.sh runs MET_MODE on two identical observations datasets. Thus, ensuring a “perfect” case. Note: The script’s initiating command requires a `start_date` and hour. The script is embedded in and called by *run_MODE_2*.

```
# Filename: run_MODE_m3o3_2_Perfect.sh
# Last Rev: 130613, gv/jr/yr
#
# Location: carson:/MET_MODE
#
# PURPOSE: Sets up and runs MET MODE on post-processed 1km res WRF
# forecast data which has been remapped to a 2.5 km grid over WSMR
# domain 3 and a 2.5 km RTMA product as the gridded observations
# dataset.
#
# Support files needed to execute this script:
#   WrfModeConfig_m3o3_rtma_WSMR_template
#
# Output includes a configuration file with the variable acronym at the
# end:
#   WrfModeConfig_m3o3_rtma_WSMR_TMP
#   WrfModeConfig_m3o3_rtma_WSMR_DPT
#   WrfModeConfig_m3o3_rtma_WSMR_WIND
#   WrfModeConfig_m3o3_rtma_WSMR_UGRD
#
# MODE results are in: /home/gvaucher/MET_MODE/results_m3o3
#
# Script called by: s_multi (select #15-Run MODE)
#
# Script OUTLINE:
#   1. User Input: Select variable.
#   2. Define variable name and variable height.
#       Script converts variable thresholds to strings for output file
# name.
#   3a. User enter object boundary definition.
#   3b. Script and User Define object boundaries.
#   4. Convert MODE configuration template into user-defined config file.
#   5. Run MODE using the user-defined specifications.
#
# =====
#
Start_Date=$1
Hour=$2

# Default is to not run the Perfect (obs vs obs) case. P=0
# Perfect case is P=1
```

This appendix appears in its original form, without editorial change.

```

P=0

mkdir -p ./results_m3o3/${Start_Date}
clear

echo "Running MODE..."
echo "  "
echo "  "
echo "  "

# *****
# 1. User Input: Select variable.
# *****

# This runs MODE on 2.5 km, WRF forecast & 2.5 km RTMA obs for testing.

echo "Enter number of task"
echo "  1 Temperature          (2m AGL)"
echo "  2 Dewpoint Temperature (2m AGL)"
echo "  3 Wind                   (10m AGL)"
echo "  4 U-component            (10m AGL)"
echo "  9 Exit"
echo "  "
echo "  "

read response1

# *****
# 2. Define variable name and variable height.
# *****

case $response1 in # Start of case
(1)
# Code for TMP.
varname="TMP"
varheight="Z2"
htstring="2"

;;

(2)
# Code for Dewpoint - DPT
varname="DPT"
varheight="Z2"
htstring="2"

;;

(3)
# Code for WIND
varname="WIND"
varheight="Z10"
htstring="10"

;;

```

```

(4)
# Code for UGRD (u-component)
varname="UGRD"
varheight="Z10"
htstring="10"
;;

(9)
# Exit the Code
echo " "
echo "Exiting MODE."
echo " "
exit 0
;;

esac #end of case

# *****
# 3a. User enter object boundary definition.
# *****

echo " "
echo "Define object boundaries:"
echo "  a. Define UPPER and LOWER variable boundaries."
echo "  b. Define boundary as all magnitudes GREATER THAN or EQUAL TO XX"
echo "  c. Define boundary as all magnitudes LESS THAN or EQUAL TO XX"
echo "  z. Exit"
echo " "

read response2

echo " "
# *****
# 3b. Script and User Define object boundaries.
# *****

case $response2 in # Start of case
(a)
# Define UPPER and LOWER variable boundaries.

# a.1 raw_thresh - Define UPPER limit value.
#   ctnumHI is "raw_thresh" in template.
#   Sets values > raw_thresh to zero.
#
echo "Enter object's HIGHEST boundary variable value (xxx.x) in kelvin or
m/s."
echo " "
echo "      This raw_thresh entry will be converted to: <=xxx.x "

read ctnumHI
echo " "

# Converts Highest Value to a string reference for the output file.
HIct=${ctnumHI/./p}

```

```

HIct="le"${HIct}

ctnumHI="<="${ctnumHI}

# a.2 conv_thresh - Define LOWER limit value.
#   ctnumLOW is "conv_thresh" in template.
#   Defines objects >= xxx.x
#   conv_thresh is set by user.
#

echo " "
echo "Enter object's LOWEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: >=xxx.x "

read ctnumLOW
echo " "

# Converts Lowest Value to a string reference for the output file.
LOWct=${ctnumLOW/./p}
LOWct="ge"${LOWct}

ctnumLOW=">="${ctnumLOW}

# a.3 conv_radius
#   crnum is "conv_radius" in template.
#   Define conv_radius as 0 (skips convolution step, since
#       raw_thresh introduced zeros).
#       For max/min extremes, conv-radius is 1 grid unit.

crnum=0

# if [ $ctnumLOW = $ctnumHI ]
#   then
#     crnum=1
# fi
;;

(b)
# Define boundary as all magnitudes GREATER THAN or EQUAL TO XX.

# b.1 raw_thresh - Define UPPER limit value.
#   ctnumHI is "raw_thresh" in template.
#   raw-thresh is set to >=-9999; to avoid false boundary data.
#   UPPER value is infinity.

ctnumHI=">=-9999"
HIct="9999"

# b.2 conv_thresh - Define LOWER limit Value.
#   ctnumLOW is "conv_thresh" in template.
#   Defines objects >= xxx.x
#   conv_thresh is set by user.

```

```

# Define LOWER value.
echo " "
echo "Enter object's LOWEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: >=xxx.x "

read ctnumLOW
echo " "

# Converts Lowest Value to a string reference for the output file.
LOWct=${ctnumLOW/./p}
LOWct="ge"${LOWct}

ctnumLOW=">=${ctnumLOW}

# b.3 conv_radius
#   crnum is "conv_radius" in template.
#   conv_radius is set to 0, which skips the smoothing step.
#   conv_radius set to 1 smooths data by 1 grid unit.
crnum=0
;;

(c)
# Define boundary as all magnitudes LESS THAN or EQUAL TO XX.

# c.1 raw_thresh - Define LOWER limit value.
#   ctnumHI is "raw_thresh" in template.
#   raw-thresh is set to >=-9999; to avoid false boundary data.
#   LOWEST value is infinity.
#
ctnumHI=">=-9999"
HIct="9999"

# c.2 conv_thresh - Define HIGHEST limit value.
#   ctnumLOW is "conv_thresh" in template.
#   Defines objects <= xxx.x
#   conv_thresh is set by user.
#
# Define HIGHEST value.
echo " "
echo "Enter object's HIGHEST boundary value (xxx.x) in kelvin or m/s."
echo " "
echo "      This conv_thresh entry will be converted to: <=xxx.x "

read ctnumLOW
echo " "

# Converts this highest Value to a string reference for output file.
LOWct=${ctnumLOW/./p}
LOWct="le"${LOWct}

ctnumLOW="<=${ctnumLOW}

# c.3 conv_radius

```



```

#     crnum is "conv_radius" in template.
#     conv_radius is set to 0, which skips the smoothing step.
#     conv_radius set to 1 smooths data by 1 grid unit.

crnum=0
;;

(z)
# Exit the Code
echo " "
echo "Exiting MODE."
echo " "
exit 0
;;

esac #end of case

clear
echo " "
echo "Run a PERFECT case? [y/n]"
echo " "
echo "    [PERFECT case is obs vs same obs data.]"
echo " "

read YN

if [ $YN = "y" ]
then
    HIct=${HIct}"_PERFECT"
    P=1
fi

echo "raw_thresh  [aka ctnumHI]  = "$ctnumHI"  "
echo "conv_radius [aka crnum]    = "$crnum"    "
echo "conv_thresh [aka ctnumLOW] = "$ctnumLOW"  "
echo " "
echo "LOWct = "$LOWct"  "
echo "HIct  = "$HIct"  "
echo " "

# *****
# 4. Convert MODE configuration template into user-defined config file.
# (Old file is deleted.)
# *****
#

echo "Removing old WrfModeConfig_m3o3_rtma_WSMR_${varname} file. Is this
ok?"
read YN

if [ $YN = "n" ]
then
    echo "Exiting script. (At least in theory...)"
    exit 0

```

```

fi

rm ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

cp ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_template
~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/varname/${varname}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>
~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/varheight/${varheight}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varna
me}1>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/ctnumLOW/${ctnumLOW}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname
}>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/ctnumHI/${ctnumHI}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/LOWct/${LOWct}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>~/ME
T_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/HIct/${HIct}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1>~/MET
_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

sed
s/crnum/${crnum}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}>~/ME
T_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1
sed
s/htstring/${htstring}/g<~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname
}1>~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}

rm ~/MET_MODE/WrfModeConfig_m3o3_rtma_WSMR_${varname}1

ls -al WrfModeConfig_m3o3_rtma_WSMR_${varname}*

# exit

# *****
# 5. Run MODE using the user-defined specifications.
# *****
#

if [ $P = "1" ]
then

mode ../MET_obs/RTMA/${Start_Date}/rtma_2p5km_${Hour}.grb1
../MET_obs/RTMA/${Start_Date}/rtma_2p5km_${Hour}.grb1
../WrfModeConfig_m3o3_rtma_WSMR_${varname} -outdir
./results_m3o3/${Start_Date} -log ./logs/test -v 2

```

```
    echo " "
    echo "PERFECT case (obs vs obs) run completed.  Exiting script."
    echo " "
    echo " "

    exit 0
fi

mode ../MET_WRFpostprd/${Start_Date}/wrf_2p5km_${Hour}.grb1
../MET_obs/RTMA/${Start_Date}/rtma_2p5km_${Hour}.grb1
./WrfModeConfig_m3o3_rtma_WSMR_${varname} -outdir
./results_m3o3/${Start_Date} -log ./logs/test -v 2

# Run completed!
echo " "
echo "<run_MODE_m3o3_2.sh> Complete."
echo " "

exit
```

List of Symbols, Abbreviations, and Acronyms

2-D	two-dimensional
AFWA	U. S. Air Force Weather Agency
AGL	above ground level
ARL	U.S. Army Research Laboratory
ASCII	American Standard Code for Information Interchange
FCST	Forecast(s)
FDDA	Four Dimensional Data Assimilation
GRIB	General Regularly-Distributed Information in Binary Form
MET	Model Evaluation Tools
MMI	Median Maximum Interest
MODE	Method for Object-Based Diagnostic Evaluation
MyWIDA	My Weather Impacts Decision Aid
NCAR	National Center for Atmospheric Research
NCEP	National Centers for Environmental Prediction
NM	New Mexico
NOAA	National Oceanic and Atmospheric Administration
NOMADS	NOAA Operational Model Archive and Distribution System
NSF	National Science Foundation
NWP	numerical weather prediction
OBS	Observation(s)
RTMA	Real-Time Mesoscale Analysis
UTC	Coordinated Universal Time
V&V	Validation and Verification
WRE-N	Weather Running Estimate – Nowcast

WRF	Weather Research and Forecasting
WRF-ARW	Weather Research and Forecasting-Advanced Research Weather Research and Forecasting
WSMR	White Sands Missile Range

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA	3 (CD)	DR J MCLAY NAVAL RESEARCH LABORATORY 7 GRACE HOPPER AVE STOP 2 MONTEREY CA 93943
2 (PDF)	DIRECTOR US ARMY RSRCH LAB RDRL CIO LL RDRL IMAL HRA RECORDS MGMT	1 (CD)	R CRAIG DAF CIVILIAN HQ AFWA 2WXG 16WS/WXN 101 NELSON DRIVE OFFUTT AFB NE 68113-1023
1 (PDF)	GOVT PRINTG OFC A MALHOTRA	3 (CD)	ARMY JOINT SUPPORT TEAM SFAE IEW&S DCGS A ATTN G BARNES 238 HARSTON ST BLDG 90060 HURLBURT FIELD FL 32544
10 (CD)	US ARMY RSRCH LAB ATTN RDRL CIE M BLDG 1622 WSMR NM 88002 B DUMAIS R FLANIGAN T JAMESON D KNAPP J PASSNER J RABY (3) B REEN J SMITH	1 (CD)	T FOWLER NCAR DEVELOPMENTAL TESTBED CENTER PO BOX 3000 BOULDER CO 80307-3000
1 (CD)	US ARMY RSRCH LAB S KIRBY ATTN RDRL CIE D BLDG 1622 WSMR NM 88002	1 (CD)	J H GOTWAY NCAR DEVELOPMENTAL TESTBED CENTER PO BOX 3000 BOULDER CO 80307-3000
12 (5 CD, 7 HC)	US ARMY RSRCH LAB G VAUCHER ATTN RDRL CIE D BLDG 1622 WSMR NM 88002	1 (CD)	J STALEY ARMY WEATHER PROPONENT OFFICE INTEGRATION SYNCHRONIZATION AND ANALYSIS (CDID) US ARMY INTELLIGENCE CENTER OF EXCELLENCE 550 CIBEQUE ST BLDG 61730 FT HUACHUCA AZ 85613
1 (CD)	US ARMY RSRCH LAB ATTN RDRL CIE BLDG 1622 WSMR NM 88002 P CLARK	1 (CD)	B BROWN JOINT NUMERICAL TESTBED PROGRAM RESEARCH APPLICATIONS LABORATORY NCAR PO BOX 3000 BOULDER CO 80307-3000
1 (CD)	US ARMY RSRCH LAB ATTN RDRL CIE D BLDG 1622 WSMR NM 88002 S O BRIEN		

INTENTIONALLY LEFT BLANK.