Optimizing and Verifying an Ensemble-based Rainfall Model

by

Sara Hargrove Friedman

B.A. Applied Mathematics and Interdisciplinary Studies Field
University of California, Berkeley, 2004

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of
Master of Science in Civil and Environmental Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2007

Author .....
Department of Civil and Environmental Engineering
May 10, 2007

Certified by .....
Dennis McLaughlin
H.M. King Bhumibol Professor of Civil and Environmental Engineering
Thesis Supervisor

Certified by .....
Dara Entekhabi
Professor of Civil and Environmental Engineering
Thesis Supervisor

Accepted by .....
Daniele Veneziano
Chairman, Departmental Committee for Graduate Students

Optimizing and Verifying an Ensemble-based Rainfall Model

by

Sara Hargrove Friedman

Submitted to the Department of Civil and Environmental Engineering
on May 10, 2007, in partial fulfillment of the
requirements for the degree of
Master of Science in Civil and Environmental Engineering

Abstract

In this thesis, I modified, optimized, and verified the stochastic Recursive Cluster-point Rainfall model of Chatdarong (2006). A novel error metric allows comparison of the stochastic ensemble of rainfall image forecasts to a single observation (radar) image. The error metric forgives position errors and provides a flexible framework for assessing how well the model works vis-à-vis a set of image measures, including the distribution of rainfall intensities over the domain at different scales.

The error metric is used in various forms to perform ad hoc optimization of the model parameters and to verify the ensemble forecast in a probabilistic framework. Verification results show that the optimized model is limited in its ability to create truly realistic rainfall patterns. Despite the model's limitations, it has unique applicability to ensemble-based rainfall data assimilation.

Thesis Supervisor: Dennis McLaughlin
Title: H.M. King Bhumibol Professor of Civil and Environmental Engineering

Thesis Supervisor: Dara Entekhabi
Title: Professor of Civil and Environmental Engineering

Acknowledgments

I am extremely grateful to my advisors, Dennis McLaughlin and Dara Entekhabi, for supporting me through many challenges and encouraging me to work on interesting problems. I have learned a lot under their guidance and thank them for the valuable experience. I must also especially thank Virat Chatdarong, whose vision and hard work laid the entire framework for my own contribution.

Thank you to my friends and colleagues from Parsons Lab for their company and kindness: Mack Durham, Pierre Gentine, Hanan Karam, Ryan Knox, Crystal Ng, Piyatida Hoisungwan, Alejandro Flores, Homero Flores Cervantes, Behnam Jafarpour, Adel Ahanin, Gautam Bisht, Rebecca Neumann, and anybody I forgot to mention. And a big thank you to Sheila Frankel, Jim Long, and Vicky Murphy for helping me navigate Parsons life!

On a personal level, I could not have accomplished this thesis without the unwavering love of Philip Hendrix and my parents. I thank them and for being there for me.

Thank you, reader. I hope you find something useful and interesting in here.

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the National Science Foundation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Need for Ensemble Rainfall Forecasts

The field of hydrology is being transformed by rapidly growing available observations and computational resources. In order to combine information from observations with multiple sources, time and space coverage, and resolutions, and to incorporate additional information from hydrological process models, the tools of data assimilation are required [7],[12]. One of the most flexible and easily implemented approaches to data assimilation is the use of ensemble-based methods.

Ensemble (or Monte Carlo) approaches approximate the full probability distribution of interest with a set of equally likely realizations (samples) of the true distribution. The set of realizations is called the ensemble, and the realizations themselves are called replicates or ensemble members. Using an ensemble, it is straightforward to approximate derived distributions by feeding each realization separately though a model. The ensemble can also be updated with measurements in standard data assimilation techniques such as the Ensemble Kalman Filter or EnKF [12]. EnKF is a recursive filtering approach that updates process model forecasts with measurements, using an approximation of the Bayesian estimation framework. For hydrological applications, we are interested in both derived distributions and data assimilation.

Rainfall ensembles are crucial for land surface hydrology applications of ensemble data assimilation. Since Eagleson's foundational work in ecohydrology [4], hydrologists have

used uncertainty in rainfall patterns as the driver of uncertainty in land surface hydrology in general. If we create an ensemble of rainfall patterns that captures the correct rainfall uncertainty, and use this ensemble as random forcing in a land surface model, we can derive probability distributions for all major surface hydrological variables of interest: runoff, soil moisture, groundwater recharge, and evapotranspiration. Such a derived distribution constitutes a prior that can be updated with measurements using data assimilation. Therefore we see that reasonable ensemble rainfall forecasts are a desirable precondition for land surface data assimilation, since we can use rainfall ensembles to derive informative prior distributions on land surface variables.

1.2 Rainfall Models

The two main types of rainfall models are physically-based meteorological models and stochastic rainfall models [2]. Physically-based models are more successful in accurately, quantitatively predicting precipitation. They are computationally expensive, grappling with the exceptionally complex cloud physics of phase change dynamics interacting with turbulent air flow, and are generally not explicitly random. For these reasons, meteorological models are not ideally suited to use for generating rainfall ensembles. Stochastic rainfall models, on the other hand, do not attempt to capture the true rainfall physics. Instead, stochastic rainfall models aim to be as simple as possible while still reproducing important rainfall patterns, e.g. statistics of spatial and/or temporal rainfall distributions. Stochastic rainfall models have an explicitly random character which makes it straightforward to generate ensembles. In addition, they are computationally simple enough to generate larger ensembles.

In this thesis, an existing stochastic rainfall model (the Recursive Cluster-point Rainfall Model of V. Chatdarong [2]) is optimized in its several free parameters and verified against radar rainfall observations. The model's ability to provide a reasonable prior ensemble for rainfall and/or land surface data assimilation is assessed based on the verification results.

1.2.1 The Recursive Cluster-Point Rainfall Model

The Recursive Cluster-point Rainfall (or RCR) model is a spatiotemporal ensemble-based stochastic model developed in its original form by Virat Chatdarong [2]. The information needed to run and verify the model was also available courtesy of Chatdarong's research effort. Chatdarong compiled and processed GOES[1] and NOWRAD[2] data from the Southern Great Plains 2004 experiment. All his processed data were available over the same spatial domain and resolution (from 25.85°N to 49.01°N latitude and 114.07°W to 90.12°W longitude, 0.5° grid spacing in each direction) and time range (June 1, 2004, 00:00 GMT to August 31, 2004, 23:00 GMT). Chatdarong derived the advection files for the model recursion from GOES images [2]. GOES, advection, and NOWRAD images are necessary to run and verify the model and were used in the work for this thesis.

The RCR model evolved from the cluster-point process rainfall model of Rodriguez and Eagleson (RE) [9]. The RE model is simple to implement. Cluster points appear randomly according to a Poisson process in 2D space. 2D Gaussian rain cells are born randomly, exponentially distributed in time, with spatial probabilities determined by superimposing 2D Gaussian clusters centered at each cluster point. See Figure 1-1.

The RCR model puts the RE model into a recursive form by assuming the Markov property holds in time, that is, only rain from the immediately previous time step is used to construct the rainfall field for the present time step. Rain from the preceding time step is advected and exponentially dissipated, while new rain is born via the cluster-point process. GOES cloud-top temperatures are used to condition the model so that rain cannot exist where there are no clouds, and new rain cannot be born when cloud-top temperature is too high. See [2] for details on how the advection fields and GOES masks were derived. Model

[1]The GOES (Geostationary Operational Environmental Satellite) system is operated by the National Oceanic and Atmospheric Administration of the United States and supplies a wealth of meteorological information for weather warnings and short-range forecasting. Two satellites in geosynchronous orbit near the United States provided cloudtop temperature images over the Southern Great Plains region for the SGP04 experiment at half-hourly time resolution and roughly 2.3 x 4 km^2 spatial resolution. See [2].

[2]NOWRAD precipitation images are a product of Weather Services International (WSI) Corporation. NOWRAD consists of improved datasets based on Doppler radar observations from the Next-Generation Weather Radar (NEXRAD) program. The algorithms used to derive NOWRAD from NEXRAD data are proprietary, but known to depend significantly on the reflectivity-rain rate relationship at the expense of other, more sophisticated adjustments. Despite these issues, NOWRAD products are thought to be reasonably accurate at large scale and are the most comprehensive rainfall product available over the continental United States. See [2].

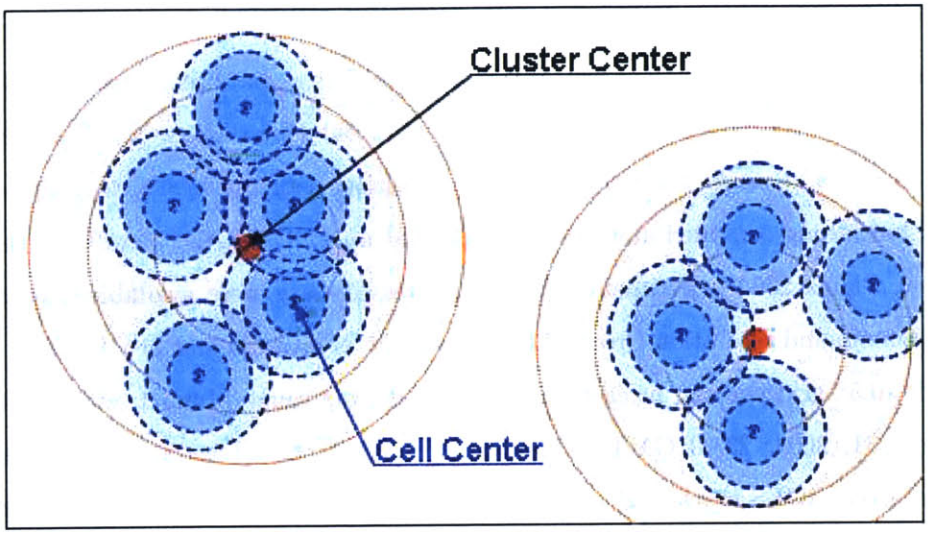

Figure 1-1: Visual representation of the probability of rain cell birth according to the cluster-point process (Chatdarong 2006).

implementation and the six free parameters are described in Appendix A of this thesis.

Chatdarong in [2] focuses primarily on rainfall data assimilation with ensembles: the RCR model is used for merging three satellite rainfall products and ground-based rain gauges, with parameters estimated through state augmentation. The model was slightly modified for use in this analysis. Please see Appendix A for a comprehensive description of the original and modified model parameters and algorithms.

It is constructive to assess and, if possible, improve the forecasting performance of the RCR model as it exists before it is used for data assimilation. Developing the best possible ensemble-based rainfall forecasting model helps provide a maximally informative prior ensemble for rainfall and land surface data assimilation. In this research effort, we calibrated (via parameter optimization) and verified the RCR model as a stand-alone ensemble forecasting model. Our conclusions highlight the successes and intrinsic limitations of the model. This analysis can aid future efforts to improve stochastic ensemble-based rainfall models and their application in hydrological data assimilation.

Chapter 2

Parameter Optimization

2.1 The Parameter Optimization Problem

The RCR parameter optimization problem is a direct global optimization in 6 continuous variables (β, ν, σ, σ_C, and $E[i_0]$), described in Table 2.1. The misfit penalty function is extremely nonlinear and not analytically known, as it is stochastic. This is thus a so-called difficult optimization problem, and furthermore is expected to have many local minima. Thus we are resigned that any feasible "optimization" will be only approximate.

Table 2.1: RCR Model Parameters

Parameter	Units	Description
β	[cells/pixel/hr]	cluster birth probability
ν	[cells/cluster]	expected cells per cluster
α	[1/hr]	exponential time decay constant
σ	[pixels]	standard deviation of Gaussian spatial spread in cell
σ_C	[pixels]	standard deviation of Gaussian spatial spread in cluster
$E[i_0]$	[mm/hr]	expected initial rain rate at cell center

The most well known heuristic for such difficult optimization problems is the simulated annealing approach [3]. Other possible tools include taboo methods and genetic algorithms.

Since the parameter space is continuous, we must also consider how to discretize the space or choose a step size for searching it.

The traditional simulated annealing algorithms use many "energy" calculations until "thermodynamic equilibrium" is reached; then the "temperature" is reduced. This is more suited to discrete combinatorial problems where the "energy" (or error) calculation is not expensive. For our problem, we wish to avoid calculating the error more than absolutely necessary. Thus we will adopt the simulated annealing approach as only a general heuristic, and design our own optimization algorithm. See Appendix B for details about the optimization algorithm and implementation.

2.2 Calculation of Error

For our application, the error calculation is not straightforward. Many judgment calls are necessary in this area as well.

To calculate the model misfit under a given set of parameters, an ensemble of rainfall images must be generated and then compared to the observed (NOWRAD) rainfall image at a given time. Thus a verification time, time allocated for spin-up, and the ensemble size must be specified a priori. The error "landscape" in the parameter space is a function of these choices – i.e. parameters will be optimized with respect to a certain verification time, spin-up time, and ensemble size.

Stochasticity of the model will play an important role, but by choosing a sufficiently large ensemble size (50-100 members) we hope to reduce the presence of noise in the error calculation. We also try, with some trial and error, to choose an optimization time for which rainfall patterns are characteristic or typical over the set of all observations. In this way we hope that optimized parameters will still perform reasonably well for other times, although we will not know whether this was successful until the model is verified. Lastly, to cut down on computation time, we use the minimum sufficient spin-up for the time decay parameter to manifest its effects (five hours).

2.2.1 Defining Error for an Ensemble Forecast

Once the ensemble of model predictions has been produced, we must compare it to a single observed rainfall image to calculate an error. Rather than a simple binary comparison of two images, we compare the observation to the entire ensemble. No given ensemble member is expected to reproduce the true image, but the ensemble as a whole should give a reasonable idea of possible rainfall patterns. Since land surface dynamics are highly nonlinear, we are not concerned with simply comparing the ensemble mean to the observation. We want to consider the ensemble as a whole, so as to examine whether the complete probability distribution sampled by the ensemble is realistic.

The observed rainfall is conceptualized as having a random nature, in that it is a single random draw from an underlying "true" probability distribution. The ensemble forecast is ideal if its members are also random draws from the same "true" probability distribution. From a statistical perspective, then, the ensemble forecast error is low if the observation is a reasonable sample from the distribution given by the ensemble. On the other hand, if the observation looks very unlikely compared to the ensemble, then the ensemble forecast has a high error.

The Mahalanobis distance of a multivariate vector $\mathbf{x} = (x_1, x_2, \cdots x_N)^T$ from a group of values with mean $\mu = (\mu_1, \mu_2, \cdots \mu_N)^T$ and covariance matrix C, is given by

$$D_{Maha.}(\mathbf{x}) = \sqrt{(\mathbf{x} - \mu)^T C^{-1} (\mathbf{x} - \mu)}$$

The Mahalanobis distance is used in statistics to detect outliers. With this approach, an ensemble gets a bigger error when the observation is more of an outlier, relative to the first two moments of the empirical distribution given by the ensemble.

The covariance matrix inversion is too computationally intensive for the full high-dimensional image, and this is why we project the image into low-dimensional space before calculating the Mahalanobis distance. The low-dimensional space is constructed with a set of image measures, to be discussed in the next section. There may be correlations and differences in variance among the measures, which is accounted for by the covariance-weighted norm taken by the Mahalanobis distance calculation.

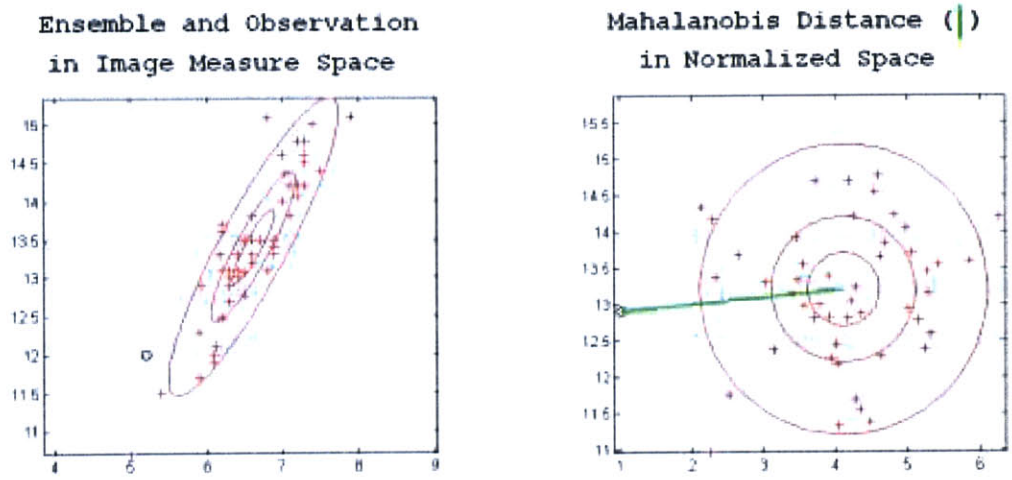

Figure 2-1: Mahalanobis distance calculation for a 2-D example. Ensemble members shown as (+), observation as (∘). Note that the distance is not calculated until the space has been normalized.

Even in the low-dimensional image measure space, problems occasionally arise with the covariance matrix inversion. A collapse to near-zero spread (a flattened covariance ellipse) in one or more directions can happen, and in these cases we face the problem of inverting a singular covariance matrix. Under a Moore-Penrose pseudo-inverse instead of a true inverse, the Mahalanobis distance calculation discards errors in the collapsed direction as zero. However, differences between the observation and the ensemble mean in a collapsed direction should be considered more, not less, significant. For this reason we do not use a pseudo-inverse. If the true inverse cannot be computed, we throw out the correlations and normalize by the variances only (a naïve interpretation of Mahalanobis distance).

Readers familiar with multidimensional ensemble verification (e.g. [5], [10]) may note the similarity with the Minimum Spanning Tree or MST, which generally involves a Mahalanobis-based distance norm. In our case we work with relatively large ensembles, which makes MST construction more demanding and suggests the more straightforward use of Mahalanobis distance. The use of spatial measures in the context of ensemble performance assessment is reminiscent of [1] and related work cited therein, but our application is different than their

statistical post-processing approach.

2.2.2 Image Measure Space

Fundamental to the error calculation procedure is the projection of the high-dimensional rainfall image into a low-dimensional space, which we call the image measure space. The N-dimensional image measure space results from applying a set of N image measures. Each rainfall image is thus reduced to an N-dimensional vector. N must be kept low in order for a reasonably sized ensemble to sample the space sufficiently. In our case of 100 replicates in the ensemble, we aim for an O(10) dimensional image measure space. Each of these N image measures should add relevant information on the nature of the rainfall and its response to changes in parameter values.

Since the model is conditioned on GOES cloudtop temperature, there is already a strong constraint on the spatial pattern of modeled rainfall, and the model parameters are structurally incapable of determining rainfall position. We are therefore indifferent to position errors, and the image measures focus on capturing the rainfall intensity distribution at both fine and coarse scales (see Figure 2-2). Multiple scales are important to consider because of the scaling properties of watershed dynamics. For the widest possible applicability, it is not only important to consider the finest scale (~50 km length scale, what we will call "1x1"), which determines local land surface response, but also the coarsest scale (~800 km length scale, or "16x16"), which determines the continental-scale land surface response including large-scale soil moisture and the discharge of major rivers. The complete rainfall intensity distributions are of interest because many land surface dynamics, such as infiltration and runoff production, are sensitive to differences in rain rate. We also consider the bulk average rain over the whole image and that fraction of pixels registering nonzero rainfall. Given the nature of the RCR model, for constant inputs the model parameters should have strong influence over such measures. The measures are also expected to vary considerably over all replicates in the ensemble, as they capture much of the stochastic effects of the model.

The following summarizes the image measures to be used for optimization. Keep in mind that the image measures to apply in general are not set in stone, but these measures are considered comprehensive and capture everything we hope the model to capture.

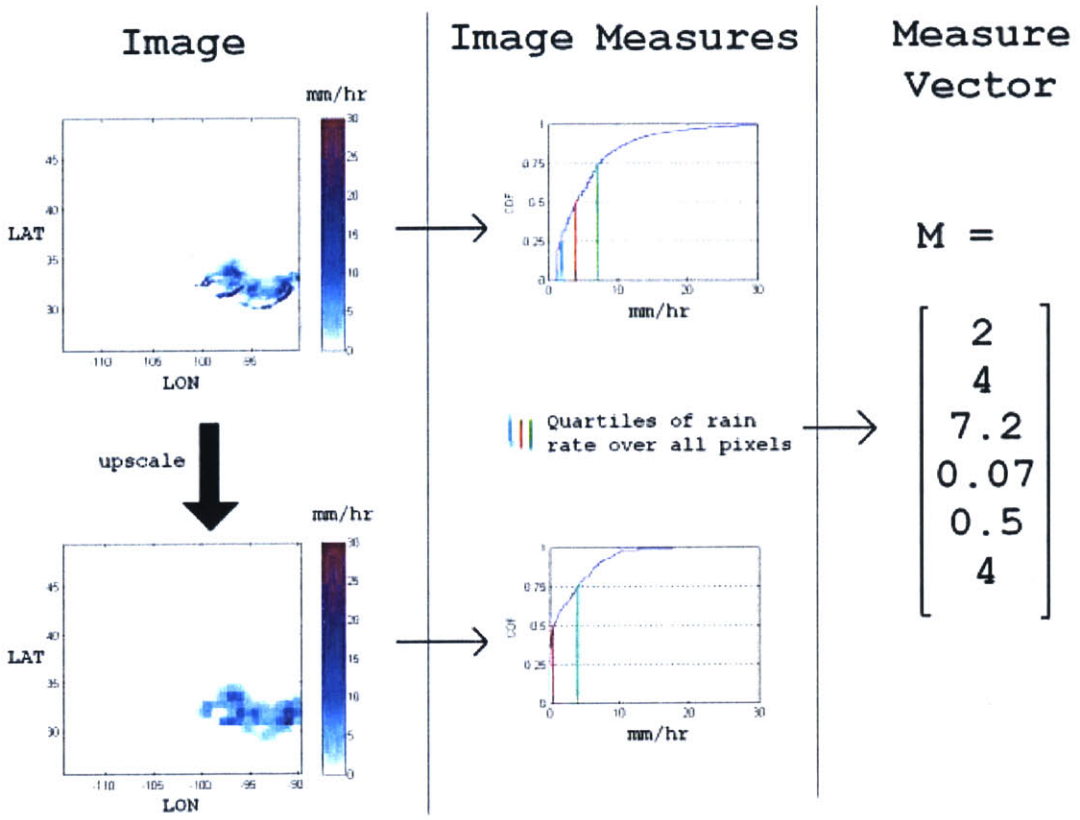

Figure 2-2: Reduction of a rainfall image (i.e. a single ensemble member or observation) into an image measure vector. In this case, the image measures are the quartiles of rain rate over all pixels at both fine and coarse scale, yielding a six-dimensional image measure vector. Actual values of the image measures are shown for the NOWRAD image in upper left corner.

For a given rainfall image, 21-dimensional measure vector M is composed of:

- Percentiles (p) 10, 20, $\cdots$ 90 of the rainfall intensity distribution over all raining ($r > 0$) pixels in the image, at both fine and coarse scales
- Average rainfall (r) over the whole image
- Fraction of pixels with nonzero rainfall, at fine, coarse scales

$$M = [\{p^{10}\ p^{20}\ \cdots\ p^{90}\}_{fine}\quad \{p^{10}\ p^{20}\ \cdots\ p^{90}\}_{coarse}\quad r_{avg}\quad f(r>0)_{fine}\quad f(r>0)_{coarse}]^T$$

2.2.3 Summary of Error Calculation Procedure

Purpose Given a set of parameter values, calculate the model error. To be called repeatedly within simulated annealing program.

- (Preliminary: Specify ensemble size, starting date-time, spin-up time)
- Using given parameters and specifications, run the model to get an ensemble of rainfall images, and open the corresponding NOWRAD observation image
- Calculate the image measure vector (M^i) for each ensemble member, $i = 1, 2 \cdots N_{reps}$, and the observation image measure vector (m)
- Calculate the Mahalanobis distance of m from $\{M^1 \cdots M^{N_{reps}}\}$

2.3 Optimization Results

The optimization program outlined previously cannot be run only once in order to yield the sole set of optimal parameters, for many reasons. First of all, the error landscape over the parameter space is defined differently for any selected optimization date-time and spin-up time, as described in the introduction to the error calculation. The error landscape may also be noisy, especially for small ensembles in an excessively high dimensional image measure space. Error noise refers to how the error may obtain different values even when run under the same specifications, because of sampling issues in the ensemble.

Even more fundamentally, the simulated annealing procedure is approximate by nature. We are not mathematically guaranteed a globally optimal solution for a continuous nonlinear problem simply by running a simulated annealing routine. In our case, this problem is exacerbated by the computational expense of the error calculation. Because error calculations are so time-consuming, we can only afford to search a very small portion of the entire six-dimensional parameter space. We hope that the simulated annealing heuristic allows us to perform the search in a useful way, and we may find a strong local minimum with our optimization program. Yet the results are necessarily imperfect in terms of finding a global minimum for the selected optimization date-time and spin-up time. When considered

together with the potential volatility of the results with respect to different optimization date-time and spin-up time, we see that a stability analysis of the optimization results is in order.

2.3.1 Optimization and Stability Analysis

For simplicity and computational feasibility, and because we do not expect it to strongly affect results, we ignore the sensitivity to spin-up time. Spin-up time will always be five hours, the minimum for significant temporal effects to manifest. We ran seven optimizations in a recursive fashion, each subsequent search beginning from the previous optimal parameters, with the first starting point for the first search in the center of the parameter space. Each subsequent search used an observation from one hour after the observation used in the previous search. In this way, we test the stability of the optimal parameters with respect to small changes in optimization date-time.

FIRST SEARCH Optimization procedure ran for 100 iterations with:

- 100 replicates
- optimization date-time 8/19/04 09:00
- initial parameters $\{\beta,\ \nu,\ \alpha,\ \sigma,\ \sigma_C,\ E[i_0]\} = \{0.005 \quad 50 \quad 1 \quad 1.5 \quad 7 \quad 5\}$ (this is roughly the center of the parameter space)

First search results are shown in Figure 2-3.

Following the first search, six more stability-testing searches were made in a recursive fashion. The stability-testing searches had a reduced computational load, with only half the iterations and half the replicates of the first search. They each started from the previous best parameters and optimized with respect to one hour after the previous date-time.

Nth SEARCH Optimization procedure ran for 50 iterations with:

- 50 replicates
- optimization date-time (N-1)th date-time + 1 hour

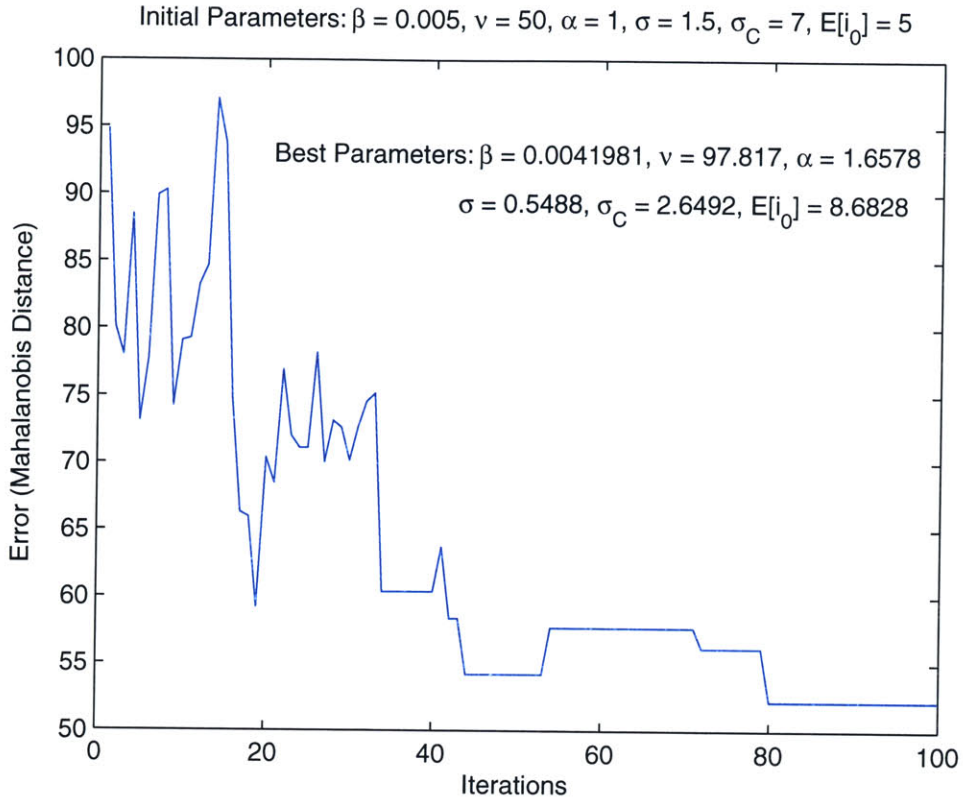

Figure 2-3: Optimization procedure results from first search.

- initial parameters $\{\beta, \nu, \alpha, \sigma, \sigma_C, E[i_0]\}$ = (N-1)th best parameters

In this way, a series of seven different sets of "optimal parameters" were found with respect to seven hourly snapshots of the storm of August 19th, 2004, from 9:00 AM to 3:00 PM. The errors of the optimal parameters with respect to their optimization date-times varied considerably, with the (apparent) best minima found in the first and second searches. See Figure 2-4.

The changes in the optimal parameters found over the seven searches are shown in Figure 2-5. The range of the y-axis in the plots is the same as the range in the parameter space.

From this, we see that there is no obvious best overall set of parameters. The optimization is quite sensitive to the specific date-time of the storm. The actual rainfall patterns are, of course, also sensitive to specific date-time. In Figure 2-6 are the NOWRAD images

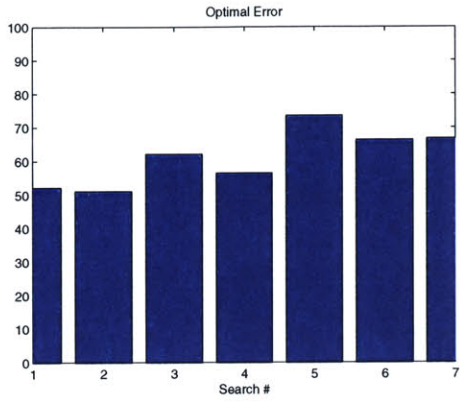

Figure 2-4: Best errors found on all seven searchs.

for the seven searches date-times. They do not appear different to the eye, but the fine and coarse scale rain rate distributions do vary significantly.

In summary, the optimal parameters found via simulated annealing are sensitive to the chosen optimization date-time. Although only minor changes resulted from re-optimizing for one hour after the first search's date-time, further stability-testing searches at later times resulted in higher best errors and significant changes in optimal parameters. This may have partly been due to the crude nature of the stability-testing searches (fewer iterations, fewer replicates). It could also have been due to significant changes in the actual rainfall patterns over time, which could have caused the error landscape to alter enough that the previous best parameters became a bad initial condition for the search. As we shall see soon, error calculation noise could be a major factor as well, since the simulated annealing program strongly rewards parameters with lowest error and fails to consider that the low error may have been just random luck of the draw.

The stability analysis results force us to conduct a final analysis to choose which set of parameters to use as our optimal parameters for the recursive cluster rainfall model.

2.3.2 Optimal Parameters: The Best of the Best

The word "best" is used in a relative sense; for this optimization problem, again, a true optimum does not exist, and even if it did it would not be findable. We do the best we can with what we have – in this case, we have the seven sets of "best parameters" found when

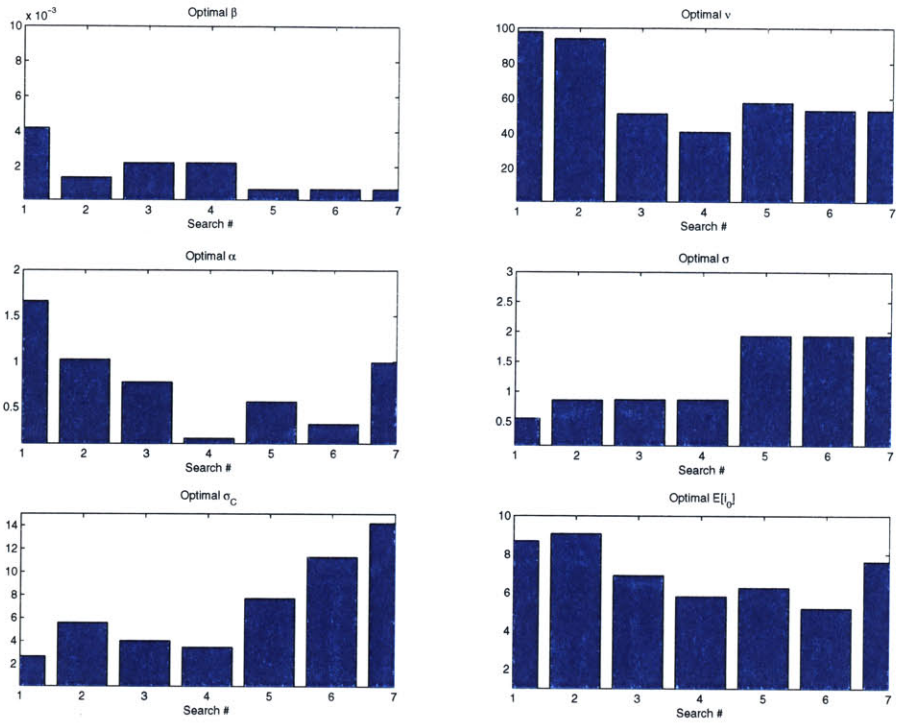

Figure 2-5: Optimal parameters found on all seven searchs. In each subplot, the range of the y-axis is the range of the parameter space.

analyzing the stability of the optimization. Each set of "best parameters" may have been the best that were found for its own optimization, but how does it fare when applied to other date-times? In the final analysis, we choose the final optimal parameters by calculating the error for each of the "best parameters" at each of the seven optimization date-times.

To reduce noise, an ensemble size of 100 was used in all error calculations. However, even with this comparatively large ensemble, the error calculation could give a much different result even when all the same conditions were specified. Thus the error calculation noise will be accounted for as well.

Error noise was investigated by simply re-calculating the error many times, using the same specifications. In this way, our stability test results give us not only the 49 errors for 7 different parameters and 7 different verification times; in fact we get 49 histograms of the error. A set of parameters that is prone to inducing collapsed ensembles and ridiculously

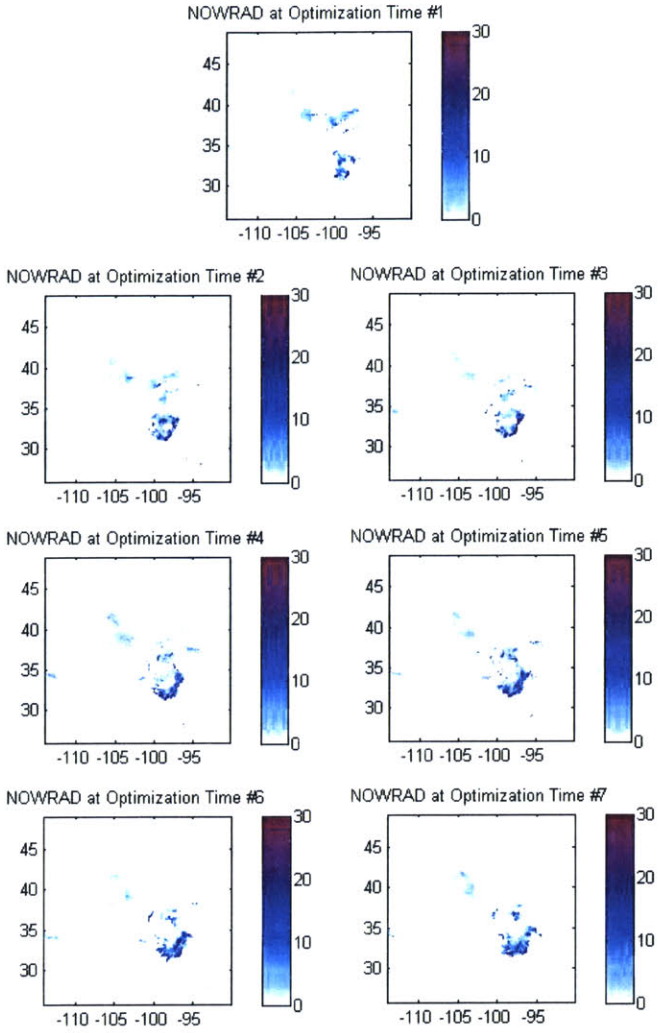

Figure 2-6: NOWRAD observed rainfall at the seven search times.

large errors will show up as a skewed histogram. We look for a set of parameters with errors values low, histograms peaky, and ridiculously high errors absent.

Fifteen samples were drawn to create each error histograms for all 49 parameter/verification time combinations. In Figure 2-7, histogram in entry (i, j) was generated for verification time j from the best parameters found in search i, denoted `PARbest(i)`. Error values were truncated to be 200 or less.

Any parameter that ever received an error over 200 or NaN, at any verification time, was eliminated from consideration. After this stage, 4 parameters remained – the first four found in the stability testing procedure. These parameters are denoted `PARbest(1)-PARbest(4)`.

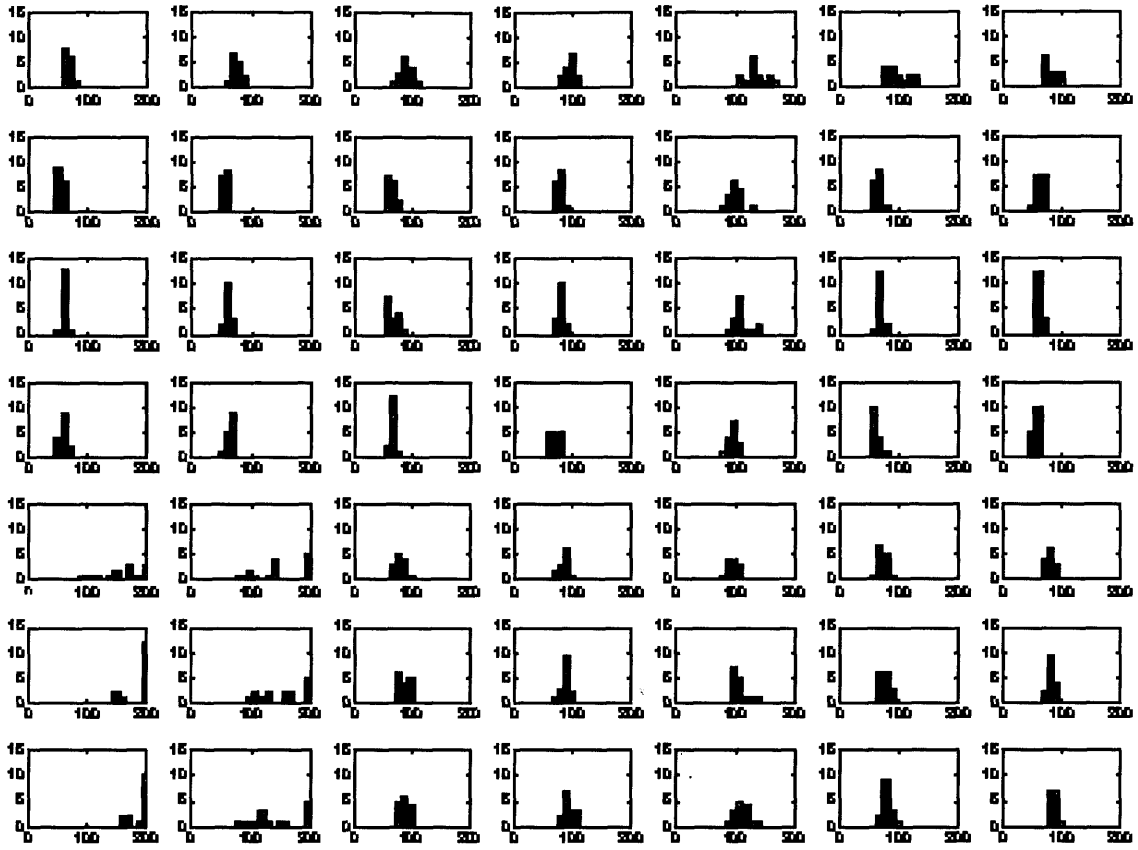

Figure 2-7: Histograms of error, `PARbest` by rows tested at verification times by column.

Several aspects of the histogram were examined in order to determine which parameters could be considered the best of the best. Ideal parameters would have low average error and low maximum error over all verification times, and their sample variances at each verification time would also be small on the average and in the maximum. These are qualities we would expect from parameters which reliably minimize error. The performance of the four `PARbests` are summarized in Table 2.2.

From this analysis, it is clear that `PARbest(4)` has the best behavior over the 7 verification times we used. It had the lowest overall errors and also had less error noise, performing best over all four measures applied.

The Best of the Best Parameters:

$\{\beta, \nu, \alpha, \sigma, \sigma_C, E[i_0]\} = \{0.0022 \quad 40.9118 \quad 0.1545 \quad 0.8678 \quad 3.4538 \quad 5.8100\}$

Table 2.2: `PARbest` errors

Parameters	Overall Average Error	Overall Maximum Error
`PARbest(1)`	92.1567	170.7072
`PARbest(2)`	69.2595	133.1677
`PARbest(3)`	73.3849	140.3905
`PARbest(4)`	**68.3436**	**110.5868**

Parameters	Average Variance of Error	Maximum Variance of Error
`PARbest(1)`	140.1880	304.3760
`PARbest(2)`	46.2115	155.1202
`PARbest(3)`	53.8807	190.5071
`PARbest(4)`	**32.4186**	**54.7825**

Chapter 3

Verification

Clearly the "Best of the Best" parameters, or indeed results from any parameter optimization for any stochastic ensemble-based high-dimensional model, cannot be trusted as truly optimal. Even if we did have optimal parameters for the model, there will be structural model error and limitations to what this simple non-physical rainfall model can predict. For these reasons, a separate model verification step must be applied to study the models forecast capabilities. In the verification analysis, the "Best of the Best" parameters were used exclusively and taken as the best possible parameters to use for all model runs.

First, to give a sense of what the model can produce for various storms, the images in Figures 3-1 – 3-10 show model output with these "best parameters" versus NOWRAD observation for five snapshots, at times selected to show a range of rainstorms over the SGP04 record. The NOWRAD observation is in the lower right-hand corner, and the other 8 images are ensemble members of the model. The date and time of the snapshot is displayed in numerical form (year month day hour, i.e. 7/25/04 at 4:00 pm is 200407251600). Latitude and longitude are shown on the axes, and colors show rainfall intensities in mm/hour. For each of the five snapshots, both fine scale raw model output and upscaled ("Coarse Scale" – each pixel averages over 16 by 16 fine scale pixels) images are shown.

Even from these few images, we can develop some intuition about the model's strengths and weaknesses. Highly scattered showers and extensive light (around 5 mm/hr) rain, such as those seen in Figure 3-5, are difficult for the model to portray. Likewise, the model cannot capture organized spatial patterns in rainstorms, perhaps the most obvious failing

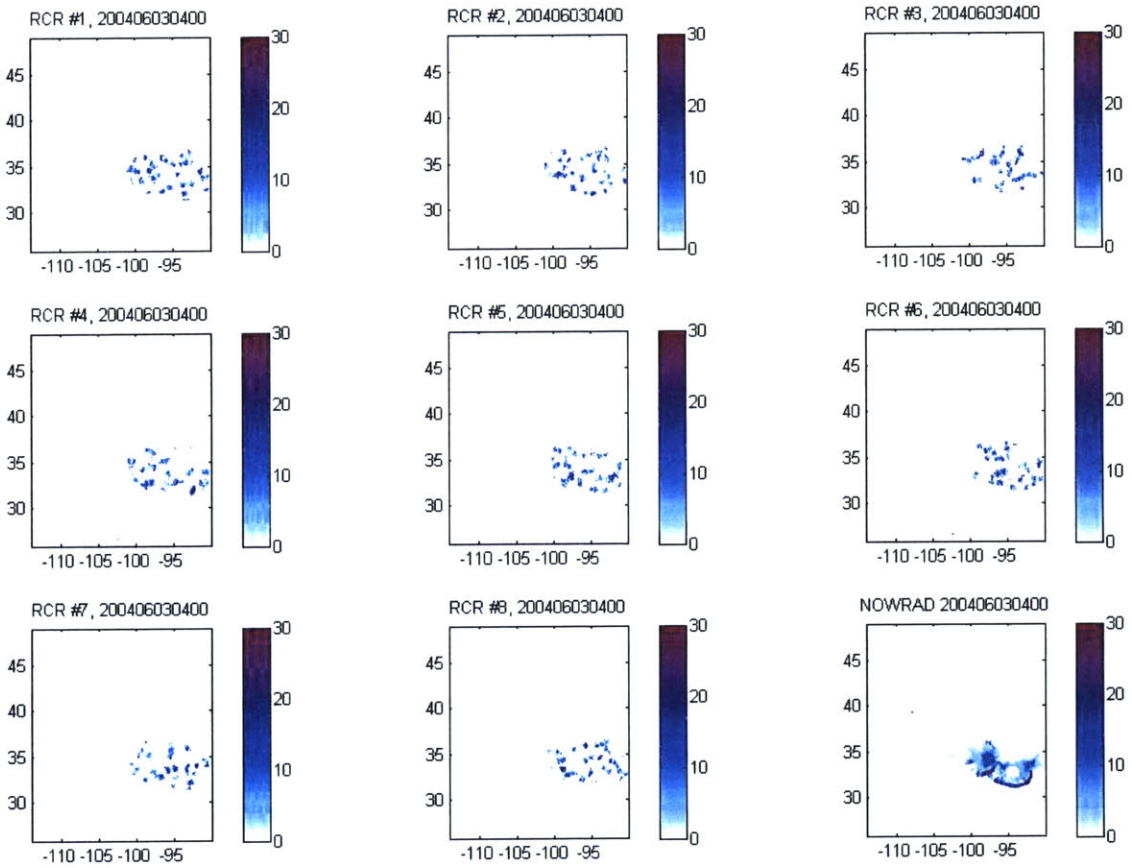

Figure 3-1: 4:00 AM, June 3, 2004

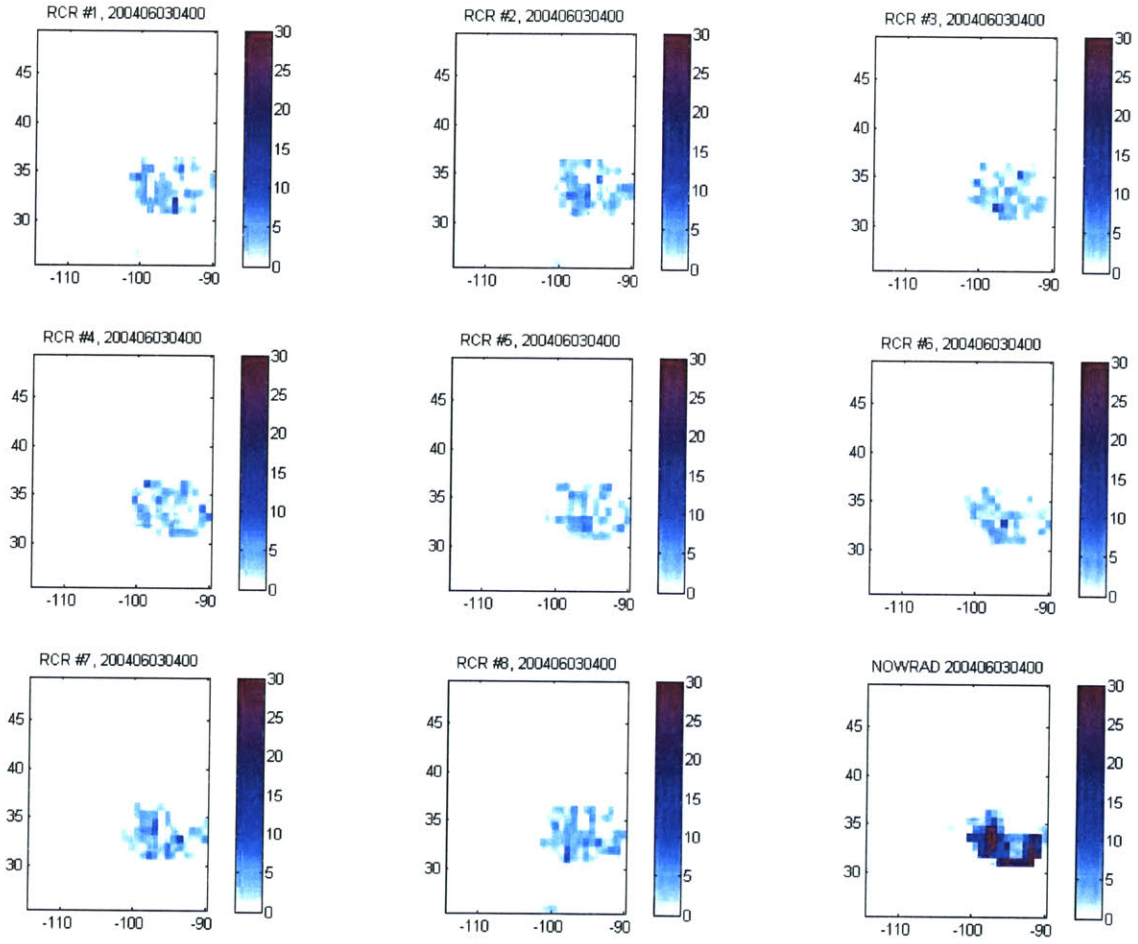

Figure 3-2: 4:00 AM, June 3, 2004: Coarse Scale

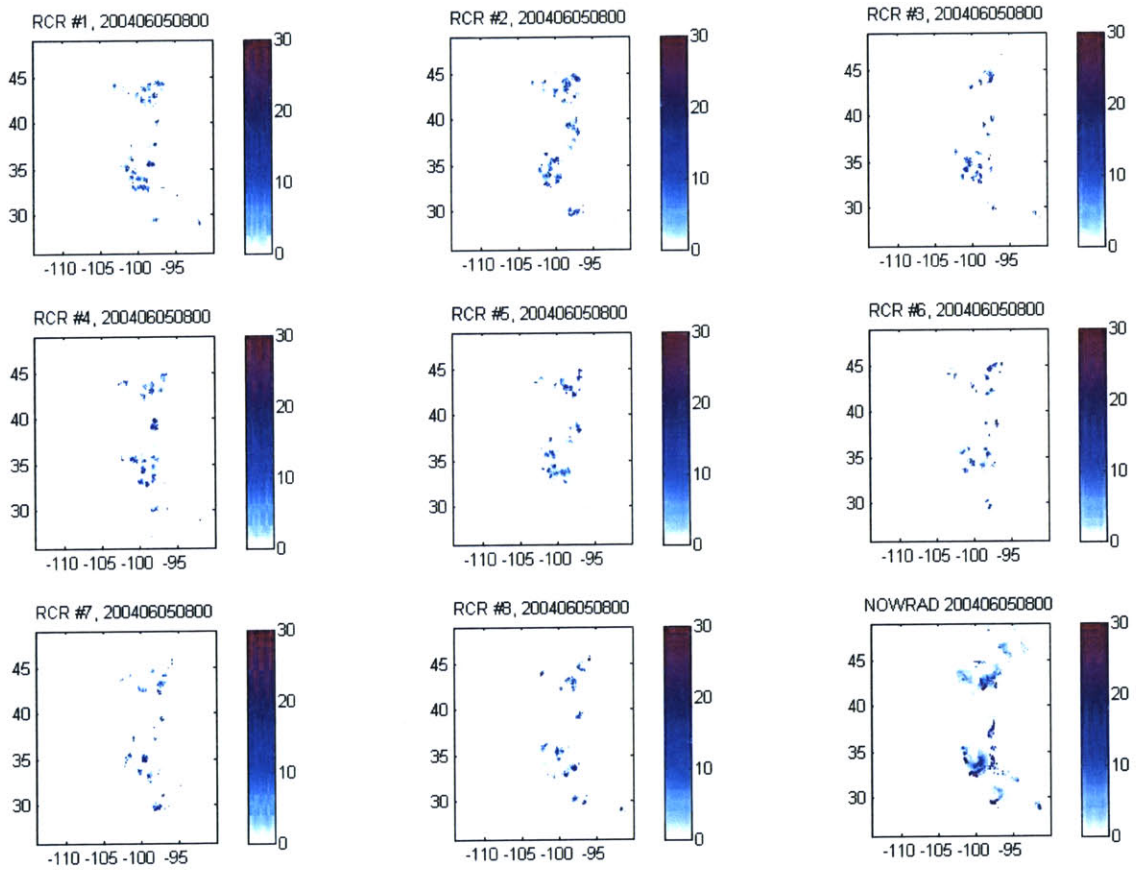

Figure 3-3: 8:00 AM, June 5, 2004

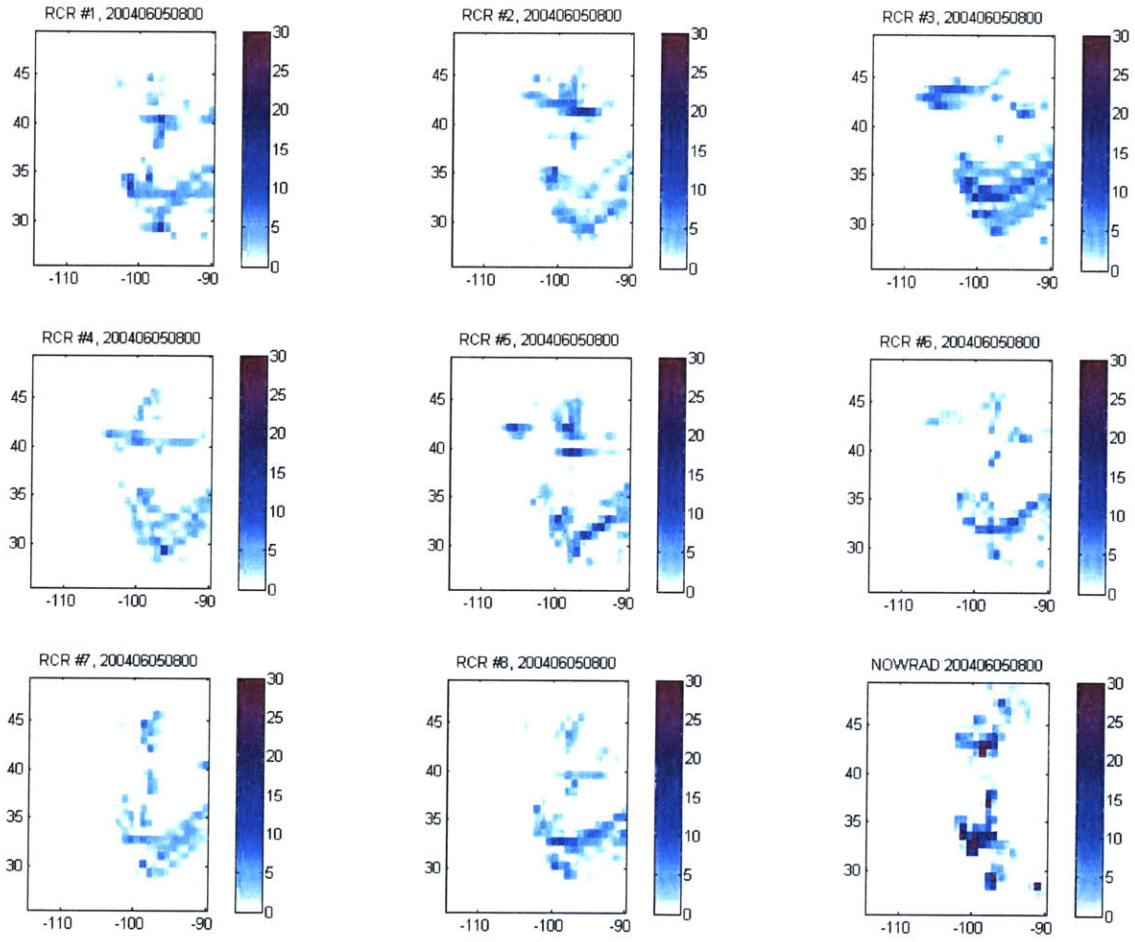

Figure 3-4: 8:00 AM, June 5, 2004: Coarse Scale

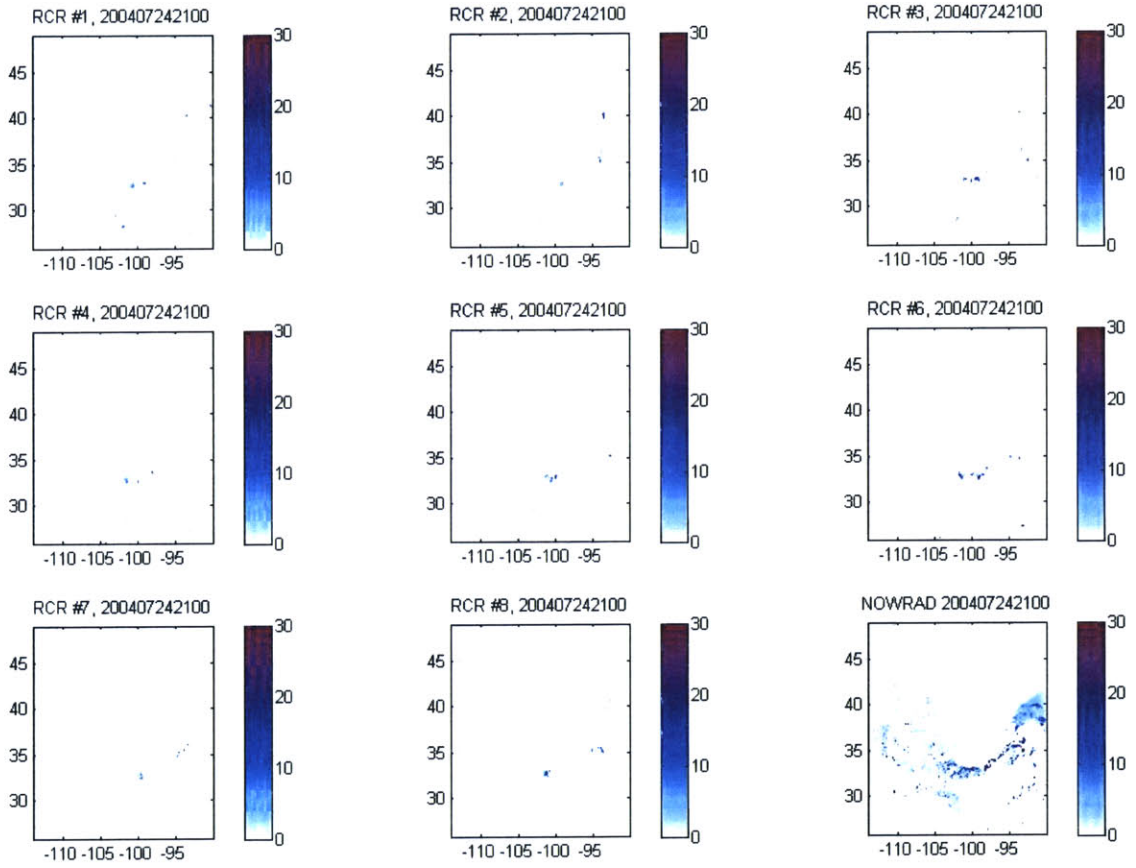

Figure 3-5: 9:00 PM, July 24, 2004

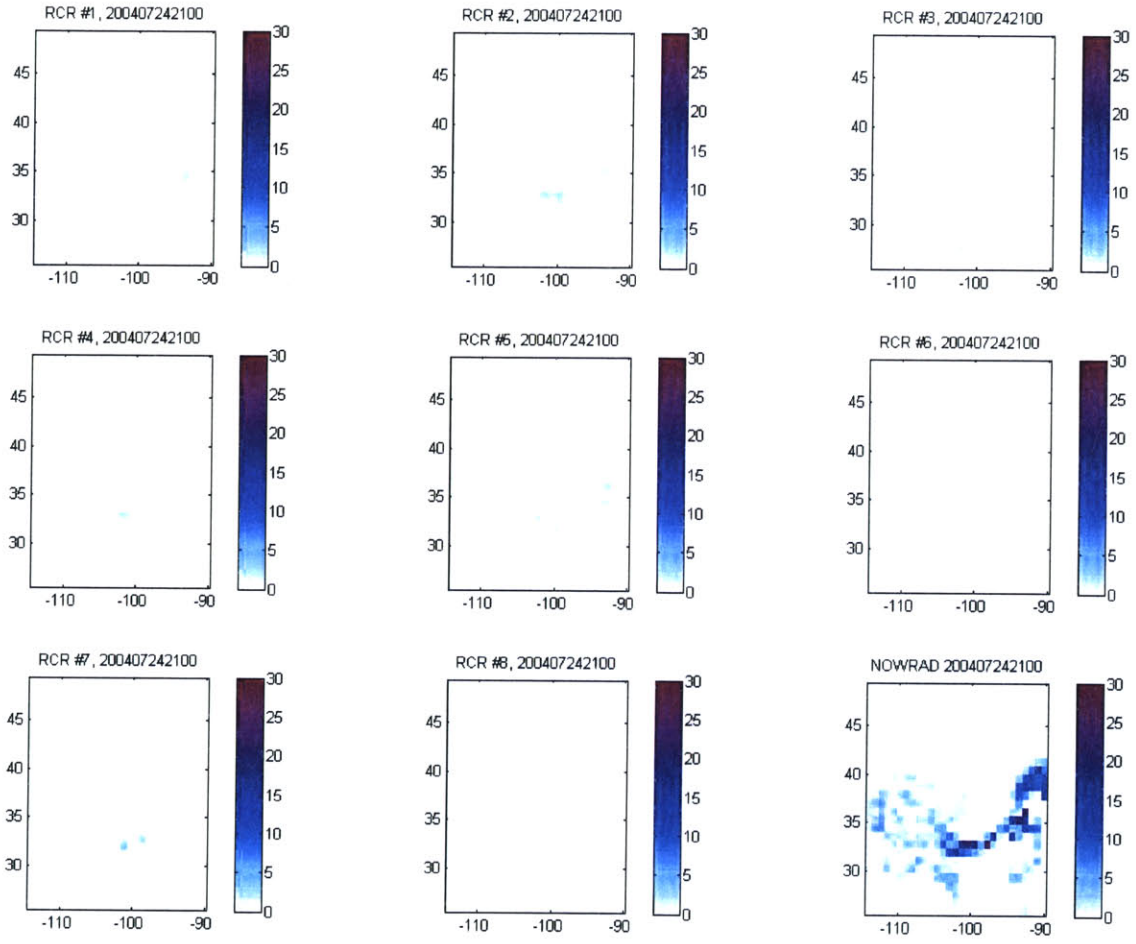

Figure 3-6: 9:00 PM, July 24, 2004: Coarse Scale

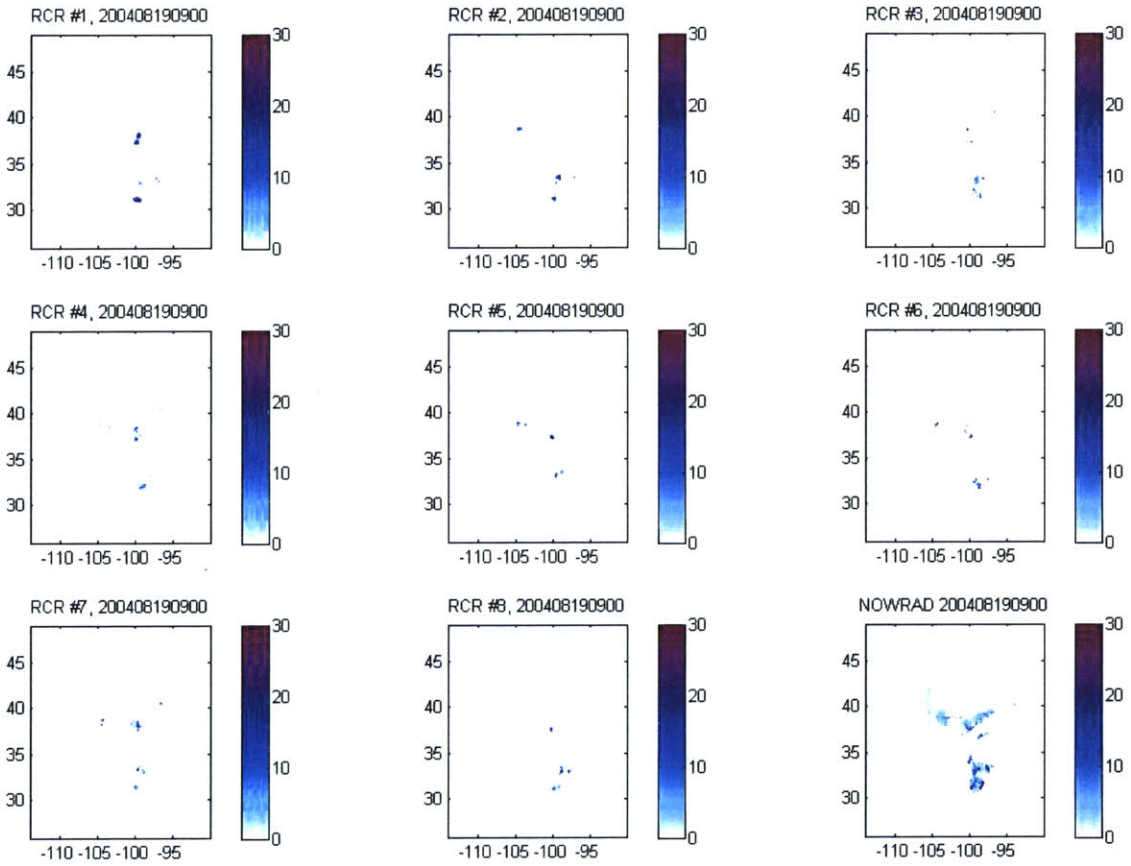

Figure 3-7: 9:00 AM, August 19, 2004

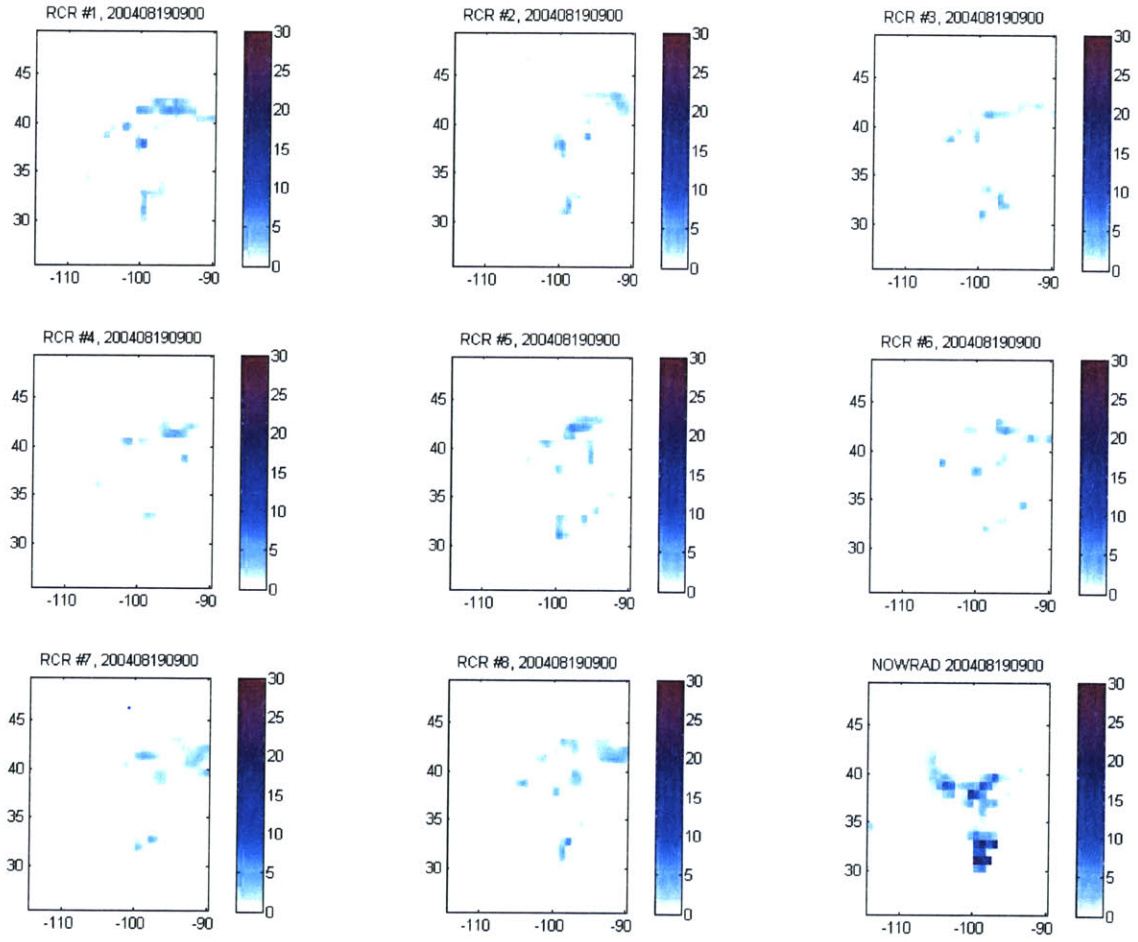

Figure 3-8: 9:00 AM, August 19, 2004: Coarse Scale

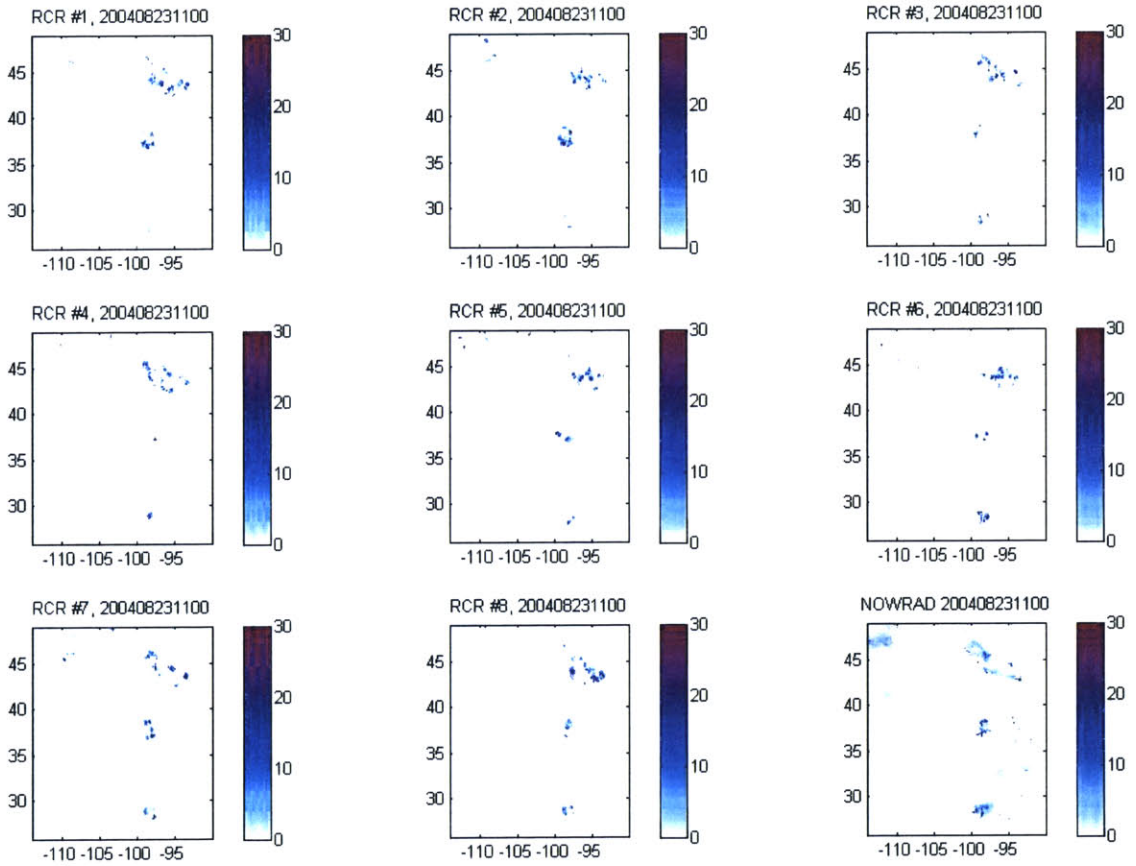

Figure 3-9: 11:00 AM, August 23, 2004

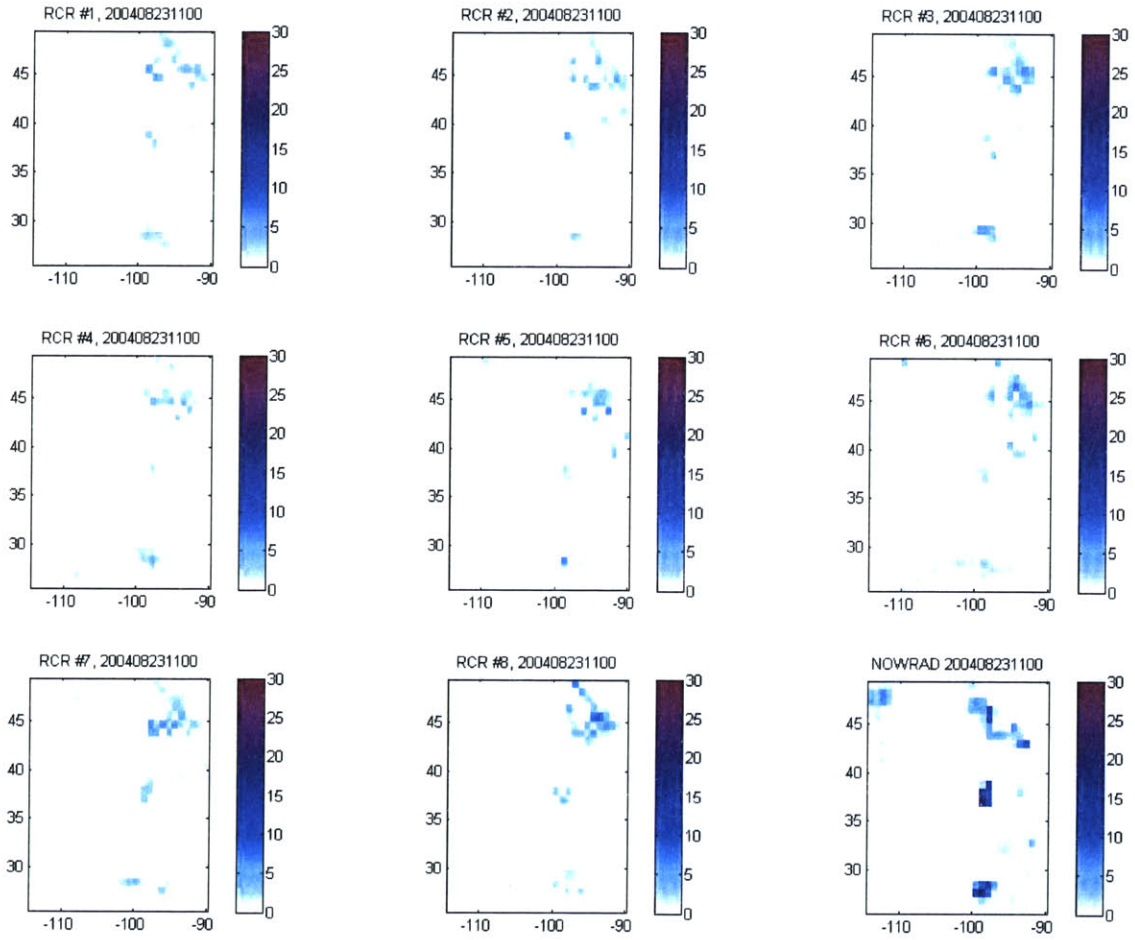

Figure 3-10: 11:00 AM, August 23, 2004: Coarse Scale

as one examines the figures by eye. The coarse scale images make clear that the model is poor at capturing both the full areal extent of rainfall as well as the extreme high rain rates found at coarse scale, the latter likely being a side effect of the modeled rain being less spatially organized into coherent bands or blobs. However, disorganized heavy rainfall, as in Figure 3-9, seems to be adequately modeled at fine scale – the NOWRAD image looks like it could possibly be another replicate of the ensemble.

Now we must be explicit about what it means for the model to be successful. Our rule of thumb is that the observation image should be distributionally indistinguishable from the ensemble members, for a good ensemble forecast. We will use the rank histogram, with our Mahalanobis-based error metric, to assess the goodness of the ensemble forecast. With our unconventional ranking strategy, we can assess both reliability and sharpness (as defined by [8]) of the ensemble forecast from the rank histogram.

3.1 Rank Histogram

The Rank Histogram is an intuitive way of testing whether, overall, the observations behave like equally likely ensemble members. An equally likely ensemble member should have a uniformly random rank over many independent verification events. A uniform rank histogram is a necessary, but not sufficient, condition for a good forecast. To determine rank, we need a scalar metric by which to order all ensemble members (plus the observation). We will use the Mahalanobis distance to order the members and observation. The Mahalanobis distance (hereafter referred to as M-distance for brevity) for each ensemble member is calculated relative to the rest of the ensemble plus the observation. The observation M-distance is calculated in the usual way, relative to the ensemble mean and covariance. If the observation is truly indistinguishable from the ensemble members, then the observation M-distance is equally likely to be ranked anywhere between 1 and N_{reps}+1 compared to the ensemble members M-distances. See Figure 3-11 for an illustration of the M-distance-based ranking strategy used to construct the M-distance-based rank histogram.

The interpretations of the M-distance-based rank histograms may be unfamiliar to readers accustomed to traditional uses of rank histograms or Talagrand diagrams for forecast

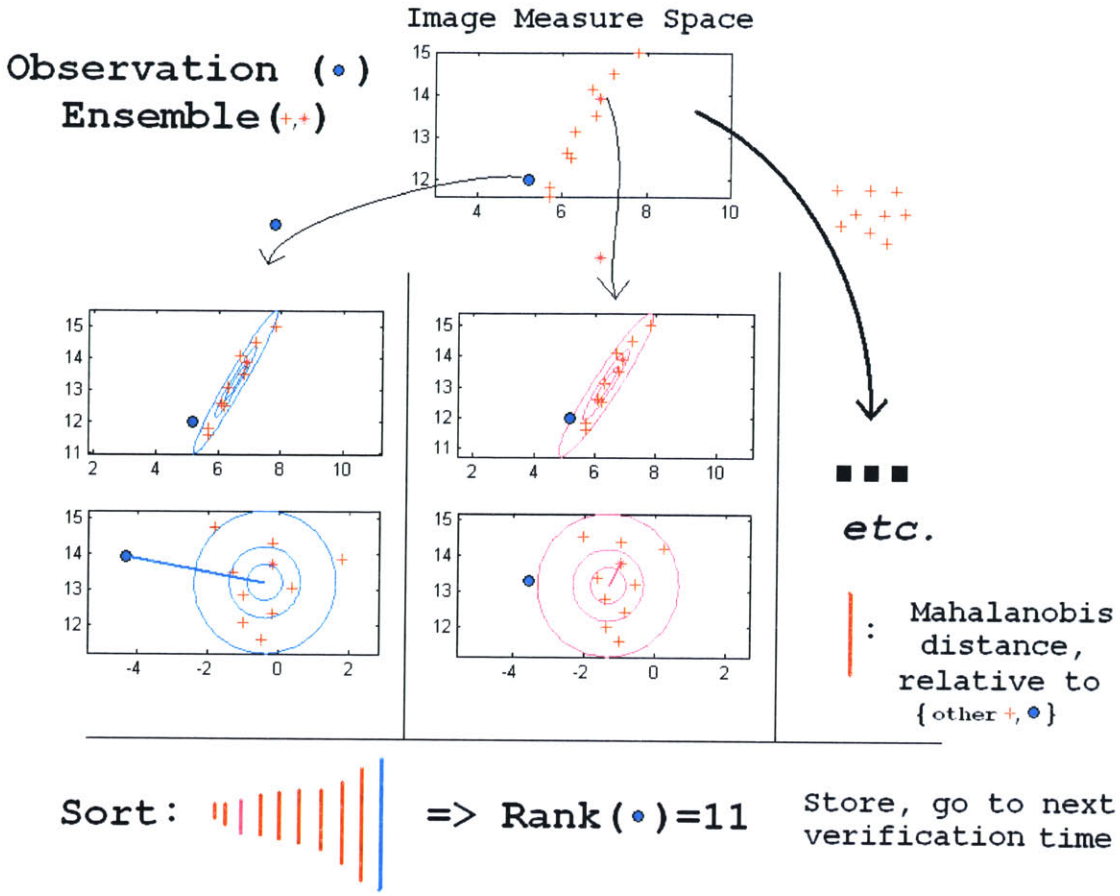

Figure 3-11: Determining the rank of the observation using Mahalanobis distance in image measure space. An augmented ensemble is created by including the observation (○). For each member of the augmented ensemble, its Mahalanobis distance is calculated relative to the rest of the augmented ensemble. This example shows M-distance calculations for the observation (○) and a selected ensemble member (∗). M-distances from the entire augmented ensemble are sorted to determine the rank of the observation.

verification (as described in [6]). Traditional rank histograms rely on a more straightforward ordering metric (e.g. wind or temperature at a single grid point) and do not contain direct information about ensemble spread. Our ranking system explicitly accounts for ensemble spread via the M-distance calculation, which affects how we interpret the rank histograms.

For our application, it makes a difference whether the rank histogram is skewed to the left or the right. A rank histogram skewed to the left does not indicate a bias; instead it indicates unbiased overdispersion or lack of sharpness in the ensemble. It means that observation M-distances are often smaller than M-distances of the ensemble members (the

observation is closer to the mean than any ensemble members). A rank histogram skewed to the right, however, indicates bias and/or underdispersion, a lack of reliability similar to that of a U-shaped histogram in a traditional application. See Figure 3-12 for an illustration.

Our implementation has many similarities to the Minimum Spanning Tree Rank Histograms of [5],[10], including the interpretation of left versus right skewness. In [10], the skewness behavior of the MST RH is examined for a synthetic example. Our implementation of Mahalanobis distance on comparatively large ensembles, however, is a more straightforward measure of the likelihood of each ensemble member, given the mean and covariance of the rest of the (augmented) ensemble. This allows enhanced interpretability of a flat rank histogram, i.e. forecast sharpness as well as reliability can be deduced in the context of our set of image measures.

It is necessary to re-examine the image measures used for the error metric. Preliminary tests determined that the error metric used for parameter optimization resulted in extremely poor rank histograms over a small set of big storms. The rank of the observation was almost always $N_{reps}+1$, indicating serious bias and/or underdispersion in the modeled ensemble. Obviously the error metric was strict. Any model failures, particularly severe underdispersion, in even a single image measure would result in a poor rank histogram – the observation would always lie so far from the ensemble, in that one dimension of the image measure space, that the observation M-distance would always be greater than any of the replicates. Using a strict error metric was appropriate for parameter optimization, since it aids the purpose of the procedure: finding parameters that improve model performance. However, we need a more flexible use of the error metric for informative model verification.

Further preliminary tests (using different image measures in the error metric) showed that the rank histogram was quite poor for certain image measures, but not for others. After the preliminary tests weeded out image measures that induced near-constant rank(obs)=$N_{reps}+1$, the following image measures remained:

- All coarse scale intensity histogram measures (p^{10}, p^{25}, p^{50}, p^{75}, p^{90} were used)
- Only the median of the fine scale intensity histogram (p^{50})

For reasons to be discussed later, the model appeared inadequate in capturing fractions

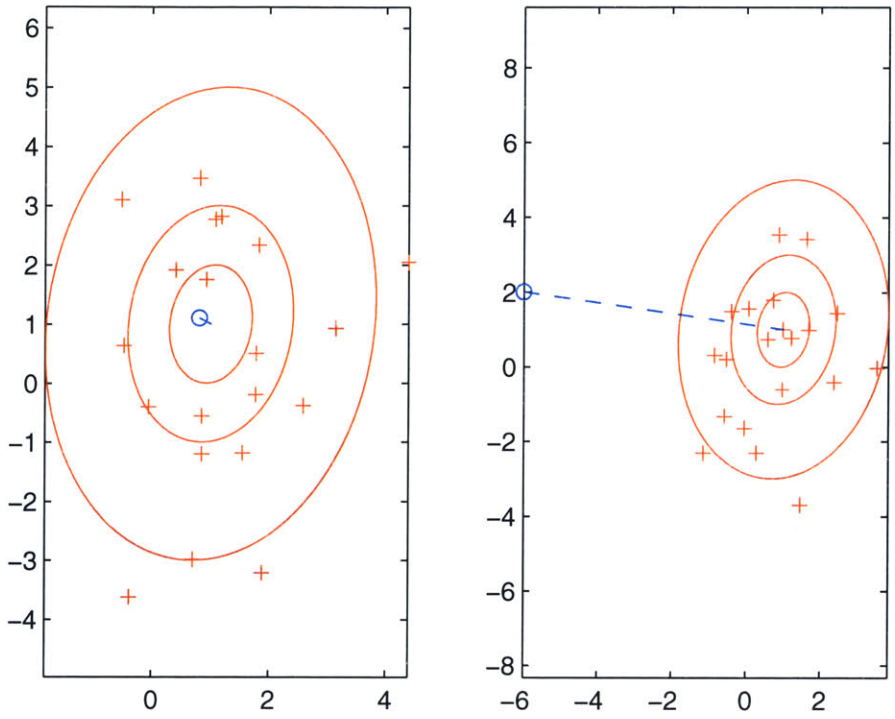

Figure 3-12: Two situations typical for different non-uniform rank histograms. LEFT: The ensemble (+) is overly dispersed and unbiased. The observation (∘) has Rank 1 under the M-distance-based ranking strategy. Note that with a different ranking strategy, i.e. sorting by x-value, the observation would have an intermediate rank. RIGHT: The ensemble (+) is underdispersed and/or biased. The observation (∘) has Rank $N_{reps} + 1$ under the M-distance-based ranking strategy. Note that with a different ranking strategy, i.e. sorting by x-value, the observation would have Rank 1. Under the M-distance-based ranking strategy, underdispersed and/or biased ensembles always get Rank $N_{reps} + 1$ and lead to right-skewed rather than U-shaped rank histograms.

of nonzero rainfall coverage (both fine and coarse scale), the average rain rate over the image, and the details of the fine-scale intensity histogram. These image measures were omitted from the new verification error metric for the rank histograms, though we revisit them later in another framework.

From visual inspection of some examples, we suspected that the model performance might be different for different types of rainfall. Verification times were then categorized into light rain and heavy rain, and separate rank histograms were constructed for each

category. We also wondered how the verifiability varied with scale, so we examined rank histograms for image measures on various upscaled images. Results are discussed below.

3.1.1 Results: Light versus Heavy Rain

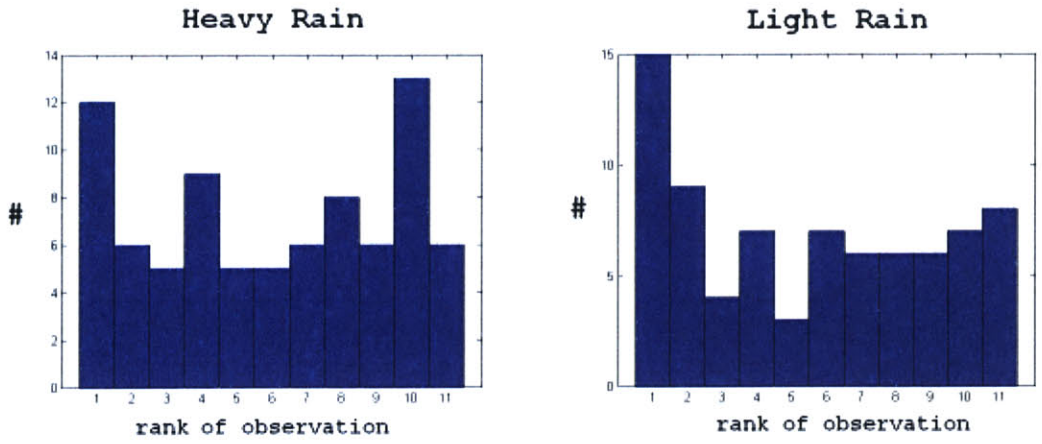

Figure 3-13: Rank histograms.

Figure 3-13 shows M-distance-based rank histograms for light versus heavy rain. Verification times were chosen where the NOWRAD images had at least 20 percent of the coarse-scale pixels raining. They were also filtered to be at least 6 hours apart in time, an ad hoc way of ensuring independence (relative to our image measures, which ignore spatial structure) across verification times. "Light rain" was defined as rain such that the NOWRAD image showed that fine-scale $p^{95} < 15$ mm/hr. "Heavy rain" was such that fine-scale $p^{95} > 15$ mm/hr. There were 79 independent light rain times, and 81 independent heavy rain times, over the SGP04 record. For each time, the model was spun up for 5 hours and then the rank of the observation stored relative to a 10 member ensemble. Since verification times were limited, the ensemble size had to be limited as well, in order for the hypothesis test of uniformity to work.

Chi-squared hypothesis testing was performed to assess whether the histograms are acceptable as random draws from the discrete uniform probability density, U[1,11]. Both histograms were accepted as uniform, with p-values 0.1347 (light rain) and 0.2800 (heavy

rain). From the p-values we can see that it is twice as likely that the heavy rain ranks are drawn from the uniform distribution, implying that the ensemble forecast is more reliable and sharper for heavy rain events.

Light rain shows a distinct spike at Rank 1. With our ranking system, this means that light rain ensemble forecasts tend to have ensemble means very close to the observation, but ensemble members far from that mean. We interpret this as an overly dispersed (not sharp enough) forecast without bias with our untraditional ranking strategy.

Now we combine light rain and heavy rain times (filtered again for 6+-hour time spacing) and examine rank histograms across scales.

3.1.2 Results: Multiple Scales

Figure 3-14 shows M-distance-based rank histograms across multiple scales. At each scale, the image measure vector was $[p^{10}\ p^{25}\ p^{50}\ p^{75}\ p^{90}]$. As we upscale the rainfall images, the rank histograms become increasingly uniform. The bias/underdispersion improves, although overdispersion becomes stronger. From this analysis, we can conclude that for reliably unbiased (though overdispersed) rainfall ensemble forecasts from the model, one must upscale, preferably by a length scale factor of 10. Such behavior (rainfall prediction quality improving as scale increases) was noted by [11], and in their terminology we could think of 10 as the "critical scale" of the RCR predictions. However, our analysis is quite different than their kriging-based methodology, and so this is only a rough correspondence.

Clearly the coarser-scale (10x10 to 16x16) rank histograms are quite non-uniform because of overdispersion. Such behavior was not apparent in the previous light versus heavy rain analysis, where the image measure vector included a sixth image measure, the fine-scale p^{50}. Perhaps underdispersion in fine-scale p^{50} somehow offset the overdispersion in the coarse-scale measures, and resulted in more uniform rank histograms. The slightly larger ensemble size of this analysis (20 instead of 10 replicates) might have more strongly identified the ensemble overdispersion as well.

We conclude the ensemble model verification by revisiting our "failed" image measures, asking the question, "how bad is bad?" We hope for insight into the precise failings of the model, which may drive future improvements.

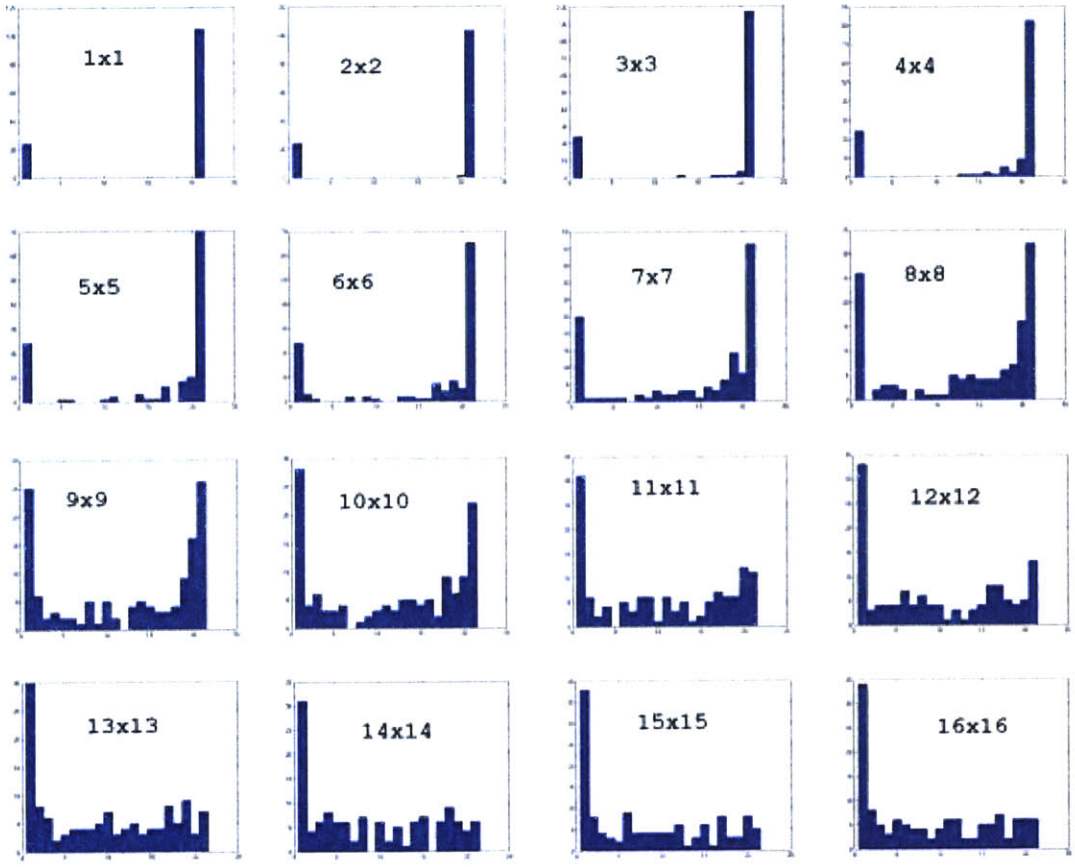

Figure 3-14: Rank histograms at scales noted. "1x1" refers to 0.5° length scale per pixel, the finest scale. Other scales are upscaled by factor shown, i.e. each "5x5" pixel would be averaged over 25 (5 by 5) of the finest scale pixels.

3.2 Assessing Outlierness

Recall that there were several measures, used for parameter optimization, which resulted in near-certain excessive M-distances of the observation relative to the ensemble. The measures incorporated most fine-scale intensity histogram measures (now encapsulated by fine-scale p^{10}, p^{90}, and IQR – the interquartile range, p^{75} minus p^{25}), the fraction of pixels raining at both fine and coarse scale, and the average rainfall over the whole image. For these six measures, we would like to answer the question, "how bad is bad?" We expect that the M-distance-based rank of the observation will tend to be N_{reps}+1, reflecting that the observation is usually an outlier in these image measure spaces. But how much of an

outlier? Is the observation just beyond the ensemble, or is it off by orders of magnitude?

To answer this question, we looked at the distribution of pair-wise distances in the ensemble (always normalized by the covariance, like the Mahalanobis distance, to create a whitened isotropic space). We then find the distribution of pair-wise distances from each ensemble member to the observation (again in the Mahalanobis-transformed space). Note that for this application of the Mahalanobis transform, the observation is taken together with the ensemble when calculating the ensemble covariance used to normalize the space. To offset sensitivity of the normalization to an outlying observation, we use a reasonably large ensemble (50 members). These observation pair-wise distances should have as much overlap with the within-ensemble pair-wise distances as possible, in order for the observation to be considered only a minor outlier. If there is no overlap at all – the observation pair-wise distances are always greater than any distances within the ensemble – we have an extreme outlier, an observation in a different galaxy.

The observation has N_{reps} pair-wise distances to the ensemble members. We can calculate the fraction of these distances which exceed the maximum within-ensemble pair-wise distance. We name this quantity the Outlierness Index, or OI.

Define:

$$D(x,y) = \text{Mahalanobis-normalized distance between } x, y$$

$$D_{max} = \text{maximum } D(x,y) \text{ with } x, y \text{ taken over all ensemble members}$$

$$\mathbf{I}[arg] = 1 \text{ if } arg \text{ is true, 0 if } arg \text{ is false}$$

Then:

$$OI = \frac{1}{N_{reps}} \cdot \sum_{i=1}^{N_{reps}} \mathbf{I}[D(obs, rep^i) > D_{max}]$$

OI is always between zero and one. An OI of one indicates an extreme outlier, and an OI less than one but greater than zero indicates an outlier that is not as extreme. In the sample calculation in Figure 3-15, OI would be zero.

The OI in the six-dimensional image measure space defined by the "bad" measures was calculated over the 128 independent rain events. The OI histogram is shown in Figure 3-16.

From these results we find that, within the image measure space of fine-scale intensity histogram measures, fractions of raining pixels, and average rain rate, the model is limited

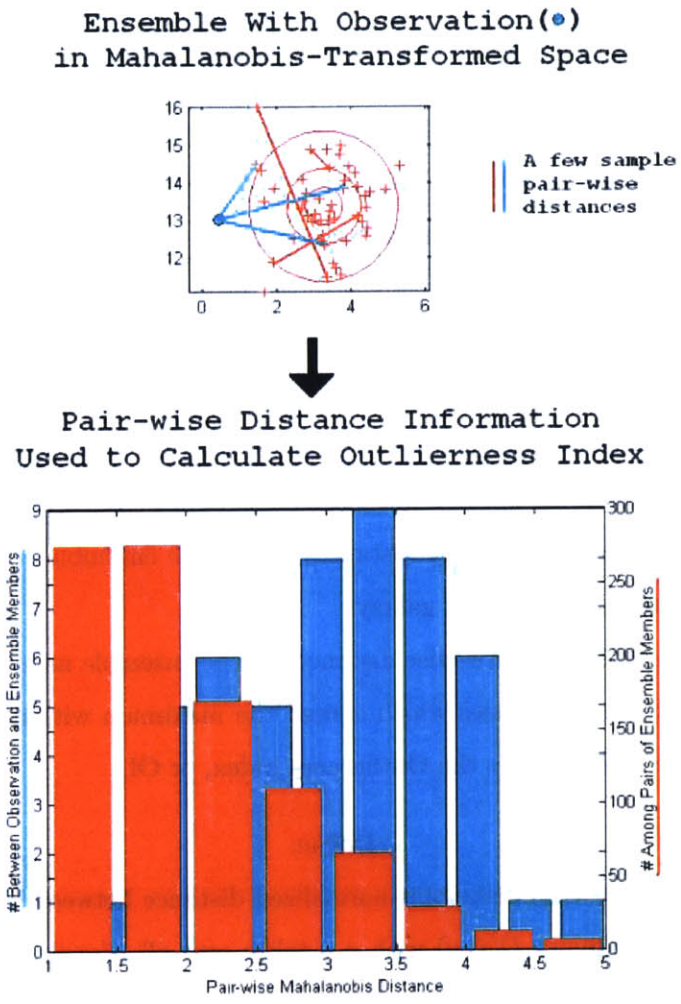

Figure 3-15: OI Calculation.

in the degree of spread it can produce. About half the time, the observation is an extreme outlier with respect to these measures. In such cases, ensembles are limited to a tight or flattened ellipsoid within this space, and the squashed ellipsoid does not generally include the "truth" (e.g. the observation). What kinds of errors does the model make to cause such common extreme outliers? See Figure 3-17 for a visualization of the ensemble versus observation in two three-dimensional image subspaces.

A sample extreme outlier is depicted in Figure 3-17. Unfortunately we cannot animate

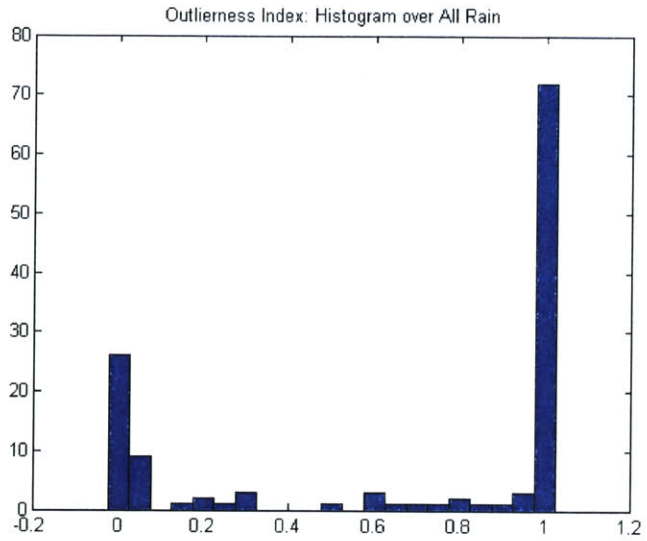

Figure 3-16: Histogram of OI over 128 independent rainfall events.

the figures to display the full 3-dimensional character, but the figures were rotated and examined within MATLAB. For this particular time, at least, the observation is actually well within the natural range of the ensemble for both the fine-scale interquartile range ($p^{75} - p^{25}$) and p^{90}, so the medium and heavy fine-scale rain is actually well captured in this case. However, the model is significantly underdispersed and biased in p^{10} (fine), r_{avg}, $f(r > 0)_{fine}$, and $f(r > 0)_{coarse}$. Overall, at this time, the ensemble members are not making rain in enough pixels, and too much extremely light (less than 0.5 mm/hr) rain.

Without looking at similar figures for all verification times, we cannot say whether the same behavior will be seen for all extreme outliers. From this single example and our understanding of the model, though, we can make some generalizations about model error. The model relies on parameters constant for all time to determine the extent and intensity of rain patterns within the cloud cover. The optimization may have overfit the model to a particular storm, and this storm of June 2 would require larger β, σ, and/or σ_C. Maybe for better behavior (more rain, and in more pixels) the parameters should vary depending on the storm character. Perhaps the GOES masks are used too bluntly, and the model is not allowed to make new rain in places where it can start raining in reality. Additionally, the model recursion uses advection and exponential decay of old rain, which creates a tendency for the model to end up with lots of raining pixels at the minimum possible rainrate (0.1

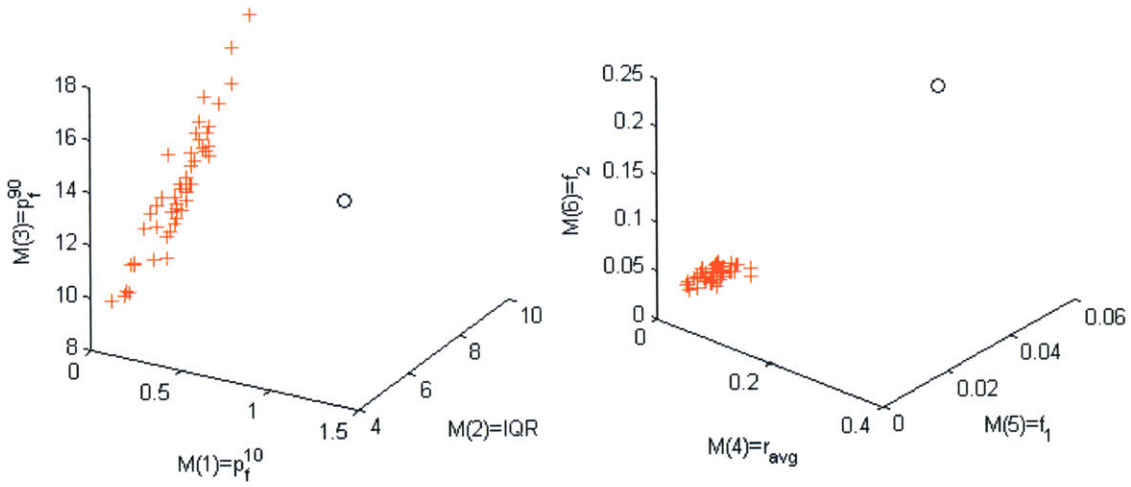

Figure 3-17: Ensemble and Observation in "bad" image measure spaces.

mm/hr). Such behavior is not physically reasonable in general and may be responsible for the underdispersion and bias in fine-scale rain rate distribution. For the June 2 case in particular, we find that in truth, the 10 percent lowest raining pixels can have intensities in excess of 1 mm/hr, whereas the ensemble is almost exclusively at fine-scale $p^{10} = 0.1$ mm/hr. So not only is there potential parametric error, but definite structural error in the model, which comes out when we examine the fine-scale rain rate distribution and spatial extent of rainfall.

Chapter 4

Conclusions and Future Research

Both calibrating and verifying the stochastic RCR model required some novel applications of the traditional tools of simulated annealing and rank histograms. Distributional and multi-dimensional measures, carefully chosen for this particular problem, were reduced to a single measure using Mahalanobis distance in the ensemble framework. The general methodology of reducing a high-dimensional ensemble to a low-dimensional measure space, then calculating the covariance-normalized distance from the observation in the low-dimensional space, is an original contribution of this research effort. The performance assessment method is flexible and could be applied to a wide range of problems dealing with uncertainty and high dimensionality.

Overall, the optimized RCR model performs reasonably well considering its simplicity and ease of use. It is generally underdispersed and biased in the distribution of fine-scale rain rates and in the spatial extent of rainfall – in the space of these image measures, the observation is an extreme outlier about half of the time, with respect to the mean and covariance of a large ensemble. On the positive side, the model is adept at characterizing the coarse-scale intensity distribution, especially during heavy rain events, which is important for land surface modeling applications.

Although the model is obviously not meant for quantitative precipitation forecasting, it is unique as a simple and computationally cheap model that can produce large Monte Carlo ensembles of random spatially and temporally distributed rainfall for hydrological data assimilation problems. When optimized using an ad hoc simulated annealing procedure

with respect to an ensemble-based error metric that forgives position errors, the model produces imperfect but useful ensemble rainfall forecasts. Future improvements to the model should focus on bringing in more position information, and address inaccuracies in rainfall spatial extent and in fine-scale rain rate, like the tendency to create too much very light rain. In the model's present form, the forecasts are useful for ensemble-based rainfall data assimilation.

Recommendations for future research:

- Improve the RCR model's tendency to create too much very light rain, i.e. by implementing a random cut-off point around 1-3 mm/hr
- Change the RCR model to bring in information on spatial structure, using one or more of the following ideas:
  - Identify rainfall positions via GOES and conditioning new rain on identified rain patches rather than simple masks
  - Have β vary with GOES brightness temperature
  - Make β a function of the change in GOES since the last time step, using heuristics derived from convective rainfall physics
  - Bring in more randomness to β and/or masks for where new rain can happen, to address model bias/underdispersion in rainfall areal coverage
- Use other image measures to assess and calibrate the RCR model, for example measures of clustering, which may allow for better calibration and verification relative to spatial characteristics of the rainfall
- Improve calibration by searching the parameter space more thoroughly and/or optimizing with respect to multiple optimization date-times; other methods to try include:
  - Implement the model with different sets of parameters for different GOES-identified types of rain (for example convective afternoon cloudbursts, frontal synoptic systems, extensive drizzle), and calibrate relative to each rain type

  - Explore and identify trade-offs between model parameters which may happen after a random "uphill" move in the search process (like if a move to reduce β – resulting in too few cluster births – were to be compensated by increasing ν and making more cells per cluster), and if found, think about a genetic algorithm instead of or in addition to the simulated annealing search

- Apply the general performance assessment methodology to other high-dimensional problems that use ensembles to capture uncertainty

Appendix A

Modifications to the Original RCR Model

Some modifications were made to the original model in order to facilitate parameter optimization and direct comparison with NOWRAD data. The original and modified models are compared below. "V.T." denotes "Virat's Thesis" and refers to the parameters used in Virats thesis. "E(n)" refers to a series of new "eyeballed" parameters that were found to give more realistic-looking ensembles, simply when comparing to NOWRAD by eye on a small sample of large storms.

A.1 Original Model

Table A.1: RCR Model Parameters

Parameter	Units	Description	V.T.	E(1)
β	[cells/pixel/hr]	"cell birth probability" (?)	0.05	1.0
ν	[cells/cluster]	expected cells per cluster	50	100
α	[1/hr]	exponential time decay constant	0.6	0.6
σ	[pixels]	standard deviation of Gaussian spatial spread in cell	1.0	0.6
ρ	dimensionless	ratio $\sigma/\sigma_{cluster}$	2.5	5.0
$E[i_0]$	[mm/hr]	expected initial rain rate at cell center	5.0	5.0

A.1.1 Pseudocode

Purpose Generate an ensemble replicate rainfall image, given an ensemble replicate rainfall image for existing rain from the previous time step, dt (time interval in hours), parameters, advection fields (U,V), and GOES masks (`newrain`, `raincanexist`)

Evolve existing rain

- Decay existing rain by factor $e^{-\alpha \cdot dt}$, with α noised by multiplicative lognormal(0, 0.1) error
- Advect existing rainfall with U,V noised by multiplicative lognormal(0, 0.1) error

Generate new rain

- Use σ, ρ to define 2D Gaussian kernels for cluster spread and cell spread
- Locate cluster centers: pixel is a cluster center when $rand(U[0,1]) < \beta/\rho \cdot dt$
- Locate cell centers:
  1. Convolve cluster centers (image of domain with all zeros except 1 = cluster center) with the cluster spread kernel.
  2. Multiply the resulting image by $dt \cdot \nu$.$\times$ `mask_newrain`.
  3. The new image is the cell birth probability image. Pixel is a cell center when $rand(U[0,1]) <$ `cell_birth_probability_in_pixel`
- Randomize rainrates for each cell center:
  1. Find decay factor by $e^{-\alpha \cdot dt}$, with dt noised by multiplicative uniform[0,1] error (uniform random birth time over the time interval)
  2. Generate random intensity from exponential distribution with mean $E[i_0]$
  3. Cell center rainrate = intensity $\times$ decay
- Find total new rain by convolving image of cell center rainrates with cell spread kernel

Total rain = (existing rain + new rain) .$\times$ `mask_raincanexist`

A.2 Modified Model

The modified model was used exclusively in all subsequent parameter optimization and verification analysis.

Table A.2: RCR Model Parameters (modified)

Parameter	Units	Description	~V.T.	~E(1)	E(2)
β	[cells/pixel/hr]	cluster birth probability	0.001	0.01	0.005
ν	[cells/cluster]	expected cells per cluster	50	100	50
α	[1/hr]	exponential time decay constant	0.6	0.6	0.6
σ	[pixels]	standard deviation of Gaussian spatial spread in cell	1.0	0.6	1.0
σ_C	[pixels]	standard deviation of Gaussian spatial spread in cluster	2.5	3.0	10
$E[i_0]$	[mm/hr]	expected initial rain rate at cell center	5.0	5.0	5.0

A.2.1 Pseudocode

Purpose Generate an ensemble replicate rainfall image, given an ensemble replicate rainfall image for existing rain from the previous time step, dt (time interval in hours), parameters, advection fields (U,V), and GOES masks (`newrain`, `raincanexist`)

NOTE: The modified code takes in and puts out rainfall images in 16-bit integer format with units [mm*10/hr], thus rainrate accuracy is 0.1 mm/hr. This agrees with the format in Chatdarong's `mat`-files for advection fields and NOWRAD images. It also results in much more efficient memory usage so large ensembles run faster. Most calculations are done with doubles in [mm/hr], but final images are scaled by 10 and cast as `int16` at the end.

Evolve existing rain

- Decay existing rain by factor $e^{-\alpha \cdot dt}$, with α noised by multiplicative lognormal(0, 0.1) error

- Advect existing rainfall with U,V noised by multiplicative lognormal(0, 0.1) error

Generate new rain

- Use σ, σ_C to define 2D Gaussian kernels for cluster spread and cell spread
- Locate cluster centers: pixel is a cluster center when $rand(U[0,1]) < \beta \cdot dt$ **[MODIFIED]**
- Locate cell centers:
    1. Convolve cluster centers (image of domain with all zeros except 1 = cluster center) with the cluster spread kernel.
    2. Multiply the resulting image by $dt \cdot \nu$.$\times$ `mask_newrain`.
    3. The new image is the cell birth probability image. Pixel is a cell center when $rand(U[0,1]) <$ `cell_birth_probability_in_pixel`
- Randomize rainrates for each cell center:
    1. Find decay factor by $e^{-\alpha \cdot dt}$, with dt noised by multiplicative uniform[0,1] error (uniform random birth time over the time interval)
    2. Generate random intensity from exponential distribution with mean $E[i_0]$
    3. Cell center rainrate = intensity $\times$ decay
- Find total new rain by convolving image of cell center rainrates with cell spread kernel

Total rain = (existing rain + new rain) .$\times$ `mask_raincanexist`

Appendix B

The Optimization Algorithm

User specifies:

- Boundaries of feasible parameter space
- Initial parameter values
- Maximum iterations of outer loop (`itmax`)
- Functions to
  - calculate error, given a vector of parameters;
  - calculate T (temperature) as a function of iteration number;
  - calculate what step size to take as a function of iteration number

Error calculation is summarized in Section 2.2.3. We use $T = F \cdot e^{-4 \cdot it/itmax}$. F is roughly equivalent to a significant change in error ($F = 5$ was used). Step size is linear in `it` and varies from 1/5 of each variables range at `it`=1 to 1/50 of the variables range at `it=itmax`.

Metropolis Acceptance Rule: Uphill move (i.e. in a direction of increased error) accepted when

$$rand[0,1] < e^{\frac{-\Delta Error}{T}}$$

Please see flowchart in Figure B-1.

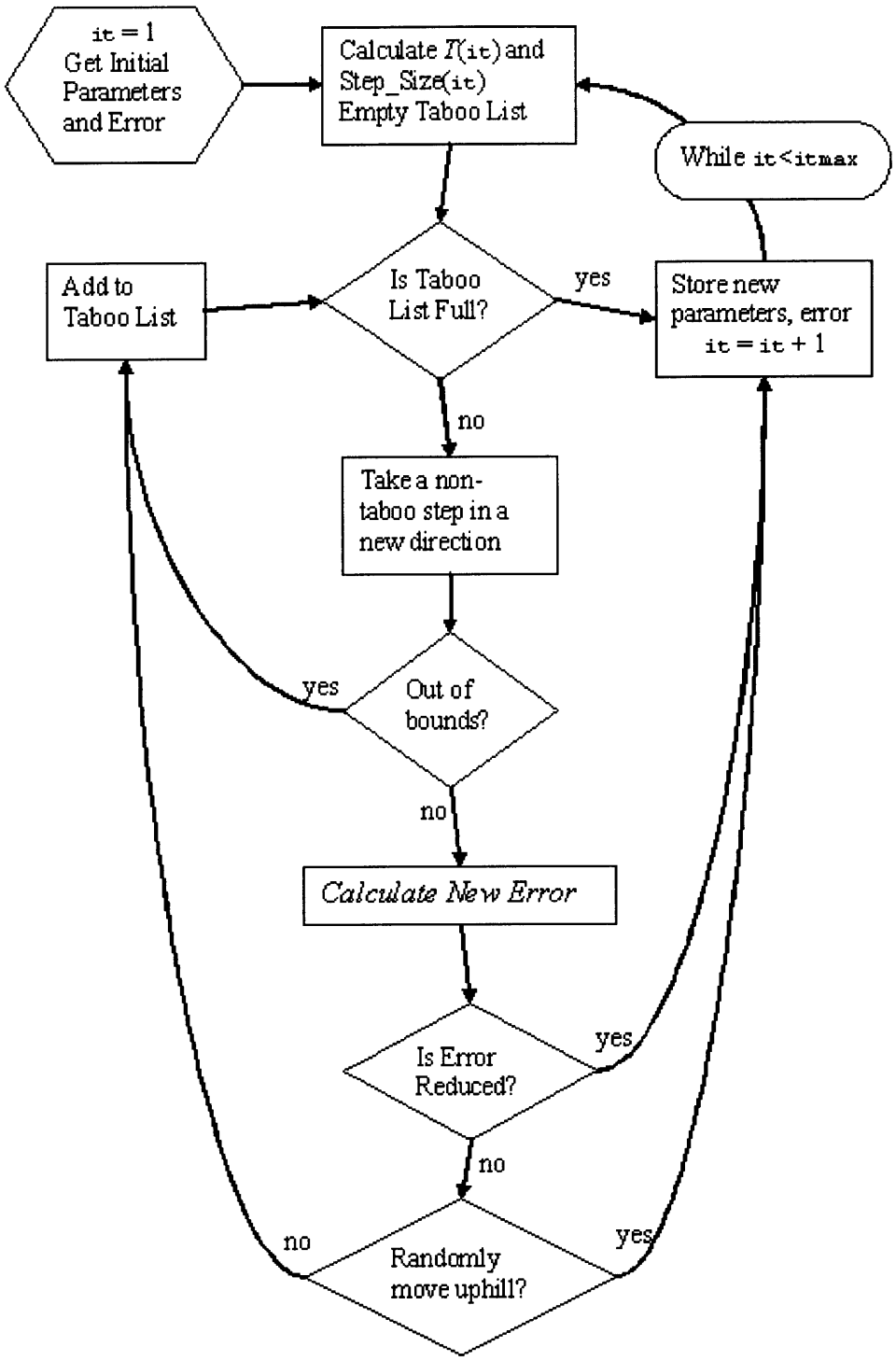

Figure B-1: The Simulated Annealing Algorithm.

Bibliography

[1] V. Berrocal, A. E. Raftery, and T. Gneiting. Combining spatial statistical and ensemble information in probabilistic weather forecasts. *Monthly Weather Review*, 135(4):1386–1402, 2007.

[2] V. Chatdarong. *Multi-Sensor Rainfall Data Assimilation using Ensemble Approaches.* PhD dissertation, Massachusetts Institute of Technology, Department of Civil and Environmental Engineering, 2006.

[3] J. Dréo, A. Pétrowski, P. Siarry, and E. Taillard. *Metaheuristics for Hard Optimization: Methods and Case Studies.* Springer, 2006.

[4] P. S. Eagleson. Climate, soil, and vegetation (7 parts). *Water Resources Research*, 14(5):705–776, 1978.

[5] D. Gombos, J. A. Hansen, J. Du, and J. McQueen. Theory and applications of the minimum spanning tree rank histogram. *Monthly Weather Review*, 135(4):1490–1505, 2007.

[6] T. M. Hamill. Interpretation of rank histograms for verifying ensemble forecasts. *Monthly Weather Review*, 129:550–560, 2001.

[7] D. McLaughlin. An integrated approach to hydrologic data assimilation: Interpolation, smoothing, and filtering. *Advances in Water Resources*, 25:1275–1286, 2002.

[8] A. H. Murphy. What is a good forecast?: An essay on the nature of goodness in weather forecasting. *Weather and Forecasting*, 8:281–293, 2003.

[9] I. Rodriguez-Iturbe and P. S. Eagleson. Mathematical models of rainstorm events in space and time. *Water Resources Research*, 23(1):181–190, 1987.

[10] D. S. Wilks. The minimum spanning tree as a verification tool for multidimensional ensemble forecasts. *Monthly Weather Review*, 132(6):1329–1340, 2004.

[11] E. Yates, J.-D. Creutin, S. Anquetin, and J. Rivoirard. A scale-dependent quality index of areal rainfall prediction. *Journal of Hydrometeorology*, 8:160–107, 2007.

[12] Y. Zhou, D. McLaughlin, and D. Entekhabi. Assessing the performance of the ensemble kalman filter for land surface data assimilation. *Monthly Weather Review*, 134(8):2128–2142, 2006.