A FLOOD NOWCASTING SYSTEM FOR THE eTHEKWINI METRO

VOLUME 1: UMGENI NOWCASTING USING RADAR – AN INTEGRATED PILOT STUDY

by

Scott Sinclair and Geoff Pegram
University of Natal

WRC Report No 1217/1/04
ISBN 1-77005-166-X

A FLOOD NOWCASTING SYSTEM FOR THE eTHEKWINI METRO

Two studies were undertaken on Flood Nowcasting for the eThekwini Metro. The first which focussed on the Umgeni River was started in early 2001 (contract K5/1217), the second was started in early 2002 (contract K8/456) and focussed on the Mlazi River. Both were completed in mid 2003. The study comprises the two projects and this Executive Summary describes each in turn. They appear in two volumes describing the two parts.

Part 1: UMGENI FLOOD NOWCASTING USING RADAR - AN INTEGRATED PILOT STUDY

EXECUTIVE SUMMARY

The original Aims of the project (K5/1217) as stated in the project proposal were:

> 1. The primary aim of this study is to provide decision makers in Umgeni Water and the Drainage and Disaster Management Departments of Durban Metro (and eventually the Umgeni Catchment Management Authority) with the tools to become proactive rather reactive in the context of flood warning.
>
> 2. To pull together the outcome of previous research funded by the Water Research Commission in the areas of Radar Estimation of Rainfall, Space Time Modelling and Forecasting of Rainfall, and Integrated Catchment and Rainfall Modelling, for Flood Forecasting in a Real World Application.

In the year 2000, the Durban Metro Disaster Management Centre did not have any facility for anticipating floods except from emergency weather reports and forecasts. They typically found themselves reacting to information phoned in by people who had either experienced damage or who had noticed that flooding was occurring. In the year 2003 they have, in the Disaster Management Centre, a GIS display overlain by real time images of rainfall measured by radar at 5 minute intervals showing them where the rain is and has fallen. They also have information coming from the Shongweni and Inanda Dams indicating the river flows, in near real time, in the two major rivers affecting the Metro.

The people living near rivers have now got the potential for some warning about impending floods and the knowledge that the Disaster Management Group is working towards mitigating floods in their area in a proactive rather than reactive way. Major industrial developments have been established in Mgeni Park and in the Mlazi Basin. Some of these are strategic industries. With the flood forecasting capability in the Durban Metro Disaster Management Centre, 6 to 12 hour warning of an impending flood will enable industry to evacuate staff and perform controlled shut downs or take steps to reduce the damage to the sensitive plants. This is a far cry from September 1987 when the

SAPREF Refinery and Mondi paper mill were closed for 10 days with serious economic consequences.

This report describes how the system which assists the Disaster Managers has been put in place. The components are meteorological, hydrological and hydraulic. The meteorological component comprises telemetering rain gauges and a radar feed from the SA Weather Service METSYS branch in Bethlehem. The hydrologic component comprises the modelling of the rainfall-runoff response of river basins from the small to the large, interpreted in a transfer function framework using a linear model developed in a previous WRC contract K5/1050: A Linear Catchment Model for Real Time Flood Forecasting. The hydraulic component comprises the work done in a subsidiary contract K8/456: Mlazi River Nowcasting to include Levels of Inundation, performed by Nokuphumulu Mkwananzi under the guidance of the Project Leader of both these contracts. This last will be reported separately in Volume 2.

Chapter 1 outlines in detail the tasks undertaken during the 2-year project which include:

- Accessing Rainfall Data: *Ensure the reliability of the Durban radar*. *Understand and write software to handle MDV format data*. *Obtain the MDV data stream directly in real-time*. *Generate catchment masks*.
- Forecasting rainfall fields and inclusion of the String of Beads Model: *Set up a forecasting version of the rainfall-runoff model*. *Set up a conditional forecasting implementation of SBM*.
- Validation and testing: *Validation on "unseen" historical data (offline)*. *Validation and testing in real-time (online)*. (This task was not completed because the real-time data were not available until after completion of the project.)

In addition to the original aims of the project (all of which were achieved except those relating to real-time data hydrological data acquisition system and its consequences, which has now been put in place in August 2003) useful work was done on rainfall forecasting and estimation. The forecasting of rainfields in real time gives good forecasts up to an hour ahead over a large area; the improved estimation comes from the merging of information from raingauges with that from radar. This work was done in place of the few undeliverable tasks.

Chapter 2 outlines the Flood Forecasting System and its components in some detail. Some of the innovations detailed there include descriptions of:

- Development of algorithms for merging of raingauge and weather radar estimates of rainfall over large areas - this provides good information in detail about where it is raining and is essential input to the decision making system
- Employment of GIS to capture real-time rainfall over sub-catchments as input to the rainfall/runoff model by integrating the optimal spatial rainfields
- Derivation of a short term rainfield nowcasting method to advect possible future rainfields in real time in order to anticipate where it will rain up to an hour ahead. This makes use of a space-time model of rainfields developed under a previously completed WRC project K5/1010 (Clothier and Pegram, 2002)
- Exploitation of the speed and efficiency of a linear transfer function rainfall/runoff model developed under a previous WRC project K5/1005 (Pegram and Sinclair, 2002) to make flood nowcasts - this was exploited in the parallel Mlazi study

- Establishment of a GIS-based information base in the eThekwini Metro Disaster Management Centre, giving instantaneous visual display of real-time information on whereabouts of storms (and their nowcasts) relative to suburbs, townships, rivers, roads etc

Chapter 3 summarises the deliverables and tasks achieved in the project,

In conclusion, the pilot project has been successful enough in its application in Durban to persuade the WRC to fund a new project K5/1429 which started in 2003: National Flood Nowcasting Initiative: Towards an Integrated Mitigation Strategy. This is being undertaken by personnel in SAWS:METSYS, DWAF & NDMC under the leadership of the Civil Engineering Programme of the University of Natal, DURBAN.

Part 2: MODELLING FLOOD INUNDATION IN THE MLAZI RIVER UNDER UNCERTAINTY

EXECUTIVE SUMMARY

The Disaster Management Group of the Durban Metro attends to a range of possible disasters including fires, large road accidents, chemical spillages, building collapses in addition to floods which are the topic of this report.

The Disaster Management personnel are not trained to understand the hydraulics and hydrology of rivers. The consequence is that to inform them that a flow of say, 1200m^3/s is going to occur at a particular part of a river in an hour's time has no meaning for them, by their own admission. What they need to know is: which part of the river water course or flood plain is going to be affected by the water so that they know who to advise, where to go, and what steps need to be taken to evacuate and prepare for disaster mitigation. The way to do this is to show these lines on a map or, even better, dynamically on a GIS representation of the area in a readily updatable form. This project forms the second part of a larger initiative reported in K5/1217: Umgeni Nowcasting using Radar - an Integrated Pilot Study. The purpose of this project was to take the rainfall measured, modelled and forecast in the parent study, convert these data into river flows computed throughout the Mlazi basin and interpret these river flows as levels of inundation in sensitive areas.

The project originated from a contract between the Durban City Engineer and the Civil Engineering consulting firm, Arcus Gibb Inc, who were contracted to provide estimates of the 20 year, 50 year and 100 year flood lines in the Mlazi River Basin. The Commission extended that study under this contract to enable the information coming from the Nowcasting study to be interpreted meaningfully by Disaster Managers. So what was originally envisaged as a planning study for the City Engineer was enriched and turned into a tool of direct and immediate use in saving life, property and mitigating flood damage.

The report describes the modelling techniques employed for the Mlazi River in the context of flood analysis and flood forecasting. These techniques are applicable to an environment where there is uncertainty due to lack of historical data input for calibration and validation purposes. The process involved the integration of GIS technology, a physically based hydrological model for flood analysis, the conceptual forecasting model for real time forecasting and the hydraulic model for computation of inundation levels. The integration of modelling techniques is better explained by summarising the process into three phases:

Phase 1 Desktop catchment modelling: A continuous physically based model (US Army Corps of Engineers' HEC-HMS Model) was set up using GIS technology. The model applied the SCS-Unit Hydrograph method for the estimation of peak discharges. Synthetic hyetographs for various recurrence intervals were used as input to the model. A sensitivity analysis was implemented and subsequently the HEC-HMS model was calibrated and peak discharges simulated. The synthetic hyetographs together with results from the HEC-HMS model were used for validation of the Mlazi Meta Model (MMM) used for real time flood forecasting.

Phase 2 Implementation of the Inundation Model: The hydraulic model (US Army Corps of Engineers' HEC-RAS) was created using a Digital Elevation Model (DEM). Field survey was conducted for the purpose of capturing the roughness coefficients and hydraulic structures, which were incorporated into the model and also for the confirmation of the terrain cross sections from the DEM. Flow data for the computation of levels of inundation were obtained from the HEC-HMS model. The levels of inundation for the natural channel of Mlazi River were simulated under the one dimensional steady state analysis whereas for the canal overbank areas simulation was conducted under unsteady state.

Phase 3 Creation of the Mlazi Meta Model (MMM): The MMM used for real time flood forecasting is a linear catchment model which consists of a semi-distributed three reservoir cell model (Pegram and Sinclair, 2002). The MMM parameters were initially adjusted using the HEC-HMS model so that it became representative of the Mlazi catchment.

This work was completed within the project timetable and now provides an additional information source to the Disaster Management Centre of the eThekwini Metro.

REPORT ON CAPACITY BUILDING

There have been two facets of Capacity Building that have sprung from this project – indirect (people being exposed to the ideas and concepts but not working on the project) and direct (those people personally involved with aspects of the project). In addition, there has been a strong component of Competency Building as a direct result of the project.

Indirect capacity building.

In the Hydrology Section of Umgeni Water, where Scott Sinclair worked during 2001 and 2002, there are two PDIs who came into contact with him and his ideas on a regular basis, because they shared an open plan office with Scott. The first was Scott's immediate superior, Percy Sithole, who was one of the original Umgeni team that got involved with this project and gave it the go-ahead. The other person was Sihle Shange, a recent graduate who worked at the desk next to Scott. They were kept abreast of the developments of the project by Scott's dissemination of information in both informal and formal ways, via reports and presentations.

The 2002 final year class of 28 Civil Engineering Students in Hydrology at the University of Natal, Durban, contained 16 PDIs (of whom 5 are women) and 2 white women. The Project leader made frequent reference to the Flood Nowcasting project in class and repeated the oral presentation given at the European Geophysical Society in Nice in April 2002 entitled "Umgeni Flood Nowcasting – an Integrated Pilot Study". (Incidentally, a second presentation arising from this study was also given in Nice which was a joint paper by Pegram, Seed and Sinclair, mostly put together by Sinclair using Seed's and Pegram's ideas – "Comparison of Methods of Short-Term Rainfield Nowcasting"). These presentations tempted students to undertake dissertations under the project leader's supervision in the second semester of 2002 and 2003.

Direct Capacity Building.

In the second semester of 2001, a female final year student, Deanne Everitt, undertook a dissertation study under the supervision of the project leader entitled "Flood Impacts: Planning and Management". This was an overview study with special focus on the Mlazi catchment in Durban, and was a direct spin-off from the present study. Deanne has now joined the Centre for Research in Environmental, Coastal and Hydrological Engineering (CRECHE) in the Civil Engineering Programme at the University of Natal as an MScEng student. In 2003, a Black student, Bahla Nkoko, undertook a dissertation on the effect of sampling density of raingauges on the estimation of the Area Reduction Factor. He made use of radar images which were collected for this project and therefore gained an introduction to the techniques of spatial rainfall measurement and estimation. He submitted his dissertation in October 2003.

Nokuphumula (Phums) Mkwananzi, who works as an Engineer with Arcus Gibb, and registered for an MScEng at Natal University under the supervision of the project leader. Phums came to the project via a WRC contract K8/456 called "Extension of Research on River Flow Nowcasting to include Levels of Inundation", with particular focus on the Mlazi river which runs between the Durban International Airport and the SAPREF refinery complex. This work is reported under Part 2 of the project and was conducted by him almost single-handedly. Phums started work on the project in January 2002 (officially on the contract in April) and completed it in June 2003. He submitted his MScEng dissertation based on the project in September 2003, 20 months after starting, which is a remarkable achievement for any post-graduate student. He graduated MScEng in December 2003.

Competency Building

Because of the nature of the Research, a number of people in Umgeni Water, Durban Metro/eThekwini Municipality, METSYS/SAWS and the University of Natal have been exposed to new ideas and potentials for ameliorating flood damages; new technology has been developed and existing technology has been improved and refined. Every individual involved has grown in competence and benefitted from the project; in the long run the wider community in the region will be beneficiaries.

It was expected that the successful completion of this project would encourage other Cities and Catchment Management Authorities to adopt the methodologies for the greater good of the nation. This has already happened in the follow-on WRC project: K5/1429: A National Flood Nowcasting Initiative: Towards an Integrated Mitigation Strategy.

Technology Transfer

The presentations made as a result of these projects include:

1. Pegram, G.G.S .and Seed, A.W., (2002). 3-Dimensional Kriging using FFT to Infill Radar Data. Oral presentation at *27th EGS Assembly*, Nice, France, April.

2. Pegram, G.G.S., Seed, A.W. and Sinclair, D.S. (2002). Comparison of Methods of Short-Term Rainfield Nowcasting. Poster presentation at *27th EGS Assembly*, Nice, France. April.

3. Sinclair, D.S., Ehret, U., Bardossy, A and Pegram, G.G.S., (2003). Comparison of Conditional and Bayesian Methods of Merging Radar & Raingauge Estimates of Rainfields, Presentation at *EGS - AGU - EUG Joint Assembly*, Nice, France, April.

4. Pegram, G.G.S., (2003). The Design and Implementation of a Real-Time Flood Forecasting System in Durban, South Africa. Poster presentation, *EGS - AGU - EUG Joint Assembly*, Nice, France, April.

5. Mkwananzi, Phums, Geoff Pegram & Scott Sinclair, (2003). Modelling Flood Inundation in the Mlazi River under uncertainty, *11th South African National Hydrology Symposium*, Technikon Port Elizabeth, September.

In yet other ways, these flood studies benefited substantially from international exchanges of knowledge. Initiatives to present data and results led to fruitful discussions and the pursuit of new ideas. In particular, Professor Geoff Pegram was active in fostering Australian and European links, as marked by the following personal invitations:

1999 - present : Invited to collaborate with the Australian Cooperative Research Centre for Catchment Hydrology
2001 - Mieyegunyah Distinguished Fellow Awardee, Melbourne University - Visiting Research Fellow (12 weeks)
2002, 2003 & 2004 - Visiting Research Fellow - Civil and Environmental Engineering Department - University of Melbourne - (8 weeks)
2002 - Keynote Speaker: 27th Hydrology and Water Resources Symposium, Melbourne, 20-23 May.
2003 - Invited to participate as rapporteur (and future full member of Steering committee) in European Union project: MUSIC / CARPE DIEM Joint Workshop with End Users, at Düsseldorf-Neuss, Germany, May 27 and 28, 2003: "CURRENT FLOOD FORECASTING PRACTICE IN EUROPE"

The knowledge gained by these interactions has benefited not only the participants in Umgeni and Mlazi Flood Nowcasting, but has already realized its potential to benefit the post-graduate students working on on-going projects which are out-growths of the Water Research Commission's investment in these projects.

ACKNOWLEDGEMENTS

The authors would like to thank a great many people who have contributed to this project during the last two years. The following organizations and individuals deserve special mention.

The Project Steering Committee:

The chairman and members of the project steering committee have been particularly helpful in making suggestions and giving constructive advice.

Dr. GC Green – Water Research Commission (Chairman)

Mr H Maaren – Water Research Commission

Mr AA Mather – eThekwini Municipality

Prof. JC Smithers – University of Natal

Mr SW Gillham – Umgeni Water

Mr J Cullis – eThekwini Municipality (Disaster management)

Mr B Mitchell – eThekwini Municipality (Disaster management)

Dr. DE Terblanche – South African Weather Service (METSYS)

Mr CG Swiegers – Department Water Affairs and Forestry (DWAF)

Mr B du Plessis – DWAF

The Water Research Commission:

The authors are indebted to the Water Research Commission for providing support for the second author and funding the project.

The South African Weather Services (METSYS):

We wish to thank the members of the METSYS Division of the South African Weather Service for their generous help in supplying data and comment. We also thank them for their willingness to install and run some of the software products emanating from this project. METSYS are also thanked for initiating the procurement of telemetering rain gauges.

Umgeni Water:

For providing support and computing resources to the second author. In particular the staff of the Planning Department are thanked for their interest in the project and assistance.

The eThekwini municipality:

The disaster management team are thanked for their enthusiasm and input; helping the authors understand some aspects of disaster management. The Water Engineering Division for providing additional support and access to telemetered rain gauge information.

The Department of Water Affairs and Forestry:

For agreeing to implement real-time gauging for the Shongweni and Inanda dams.

UMGENI NOWCASTING USING RADAR – AN INTEGRATED PILOT STUDYI

CHAPTER 1.

CHAPTER 2.

CHAPTER 3.

APPENDICES

TABLE OF FIGURES

Glossary of terms

AGU	American Geophysical Union
DEM	Digital Elevation Model
DM	Disaster Manager
DWAF	Department of Water Affairs and Forestry
EGS	European Geophysical Society
EUG	European Union of Geosciences
FLOPS	Floating point operations
FTP	File Transfer Protocol
GCM	Global Circulation Model
GIS	Geographical Information System
GNU	GNU's not Unix
HEC-RAS	Hydrologic Engineering Center – River Analysis System
HTML	Hypertext Markup Language
IMF	Image Mean Flux
MDV	Meteorological Data Volume
METSYS	Meteorological Systems and Technology
NDMC	National Disaster Management Centre.
OO	Object Oriented.
RMSSE	Root mean sum of squared errors
SAWS	South African Weather Service.
SBM	String of Beads Model
WRC	Water Research Commission

CHAPTER 1 INTRODUCTION

This introductory chapter restates the original objectives of the project, and then provides some background to the project as well as describing the requirements for an effective Flood forecasting system.

1.1 - BACKGROUND

Floods cannot be prevented, but their devastating effects can be minimized if advance warning of the event is available. The white paper on disaster management (WPDM, 1998) presents Table 1.1.1 showing the estimated losses due to some recent disasters in South Africa. Of particular significance is the prominence of flood events among these disasters.

Place	Disaster	Cost
Ladysmith	Floods, 1994	• 400 families evacuated • R50 million damages
Merriespruit	Slimes dam, 1994	• 17 lives lost • R45 million damages
Pietermaritzburg	Floods, 1995	• 173 lives lost • Emergency shelter needed for 5 500
Ladysmith	Floods, 1996	Damages to infrastructure: R25 million
South Africa	Drought, 1991-92	• 49 000 agricultural jobs lost • 20 000 non-agricultural jobs lost • Associated with 27% decline in agricultural gross domestic product
Northern Province	Floods, 1996	R105 million damages
Mpumalanga	Floods, 1996	R500 million damages

Table 1.1.1 Counting the cost of some recent disasters (after WPDM, 1998)

With large increases in population and increasing urbanisation (mainly driven by poverty) there are more people living in informal settlements, which are often on floodplains as this is the only undeveloped land available near cities. The people living in these settlements are those who are most at risk, not only due to their geographical location in the floodplain but also because (without insurance policies) they do not have the financial resources to recover from the damage caused by flooding.

The newly promulgated Disaster Management Act (Act 57 of 2002) advocates a paradigm shift from the current “bucket and blanket brigade” response-based mindset to one where disaster prevention or mitigation are the preferred options. It is in the context of mitigating the effects of flood events that the development and implementation of a reliable flood forecasting system has major significance. Even a small amount of lead time, a few hours, can allow disaster managers to take steps which may significantly reduce loss of life and damage to property.

1.2 - REQUIREMENTS FOR AN EFFECTIVE FLOOD FORECASTING SYSTEM

The requirements of an effective flood forecasting system need to be set out based on a concrete definition of what constitutes such a system. In order to establish a definition the members of the project team were consulted during the project proposal process and each asked to provide a summary of what they expected from the proposed flood forecasting system. The following comments are based on the original proposal made to the Water Research Commission (WRC).

The eThekwini municipality expected a system which:

- is quick and easy to run
- is easy to obtain information from
- is easy to update
- is spatially presented (preferably with a GIS)
- is able to provide predictions with a good degree of confidence
- provides reasonable advance warning of the event (2-3 hours)

Umgeni Water expected a model or suite of models that:

- use radar data as precipitation input
- forecast storm movement and precipitation
- determine the expected runoff from catchments
- are easily updateable
- have a user friendly front end

The list of requirements above was used to guide the design of the system in conjunction with current practice found in the literature.

Of key importance are reliable real-time measurements of rainfall and streamflow. It is usually difficult to get people who are in danger to move away from their homes and possessions therefore it is of the utmost importance that the warnings are reliable to facilitate the development of a relationship of trust between the affected communities and the Disaster Managers. There is a very high cost associated with “crying wolf”, i.e. false alarms.

Some of the requirements had to be balanced with practical requirements due to the limited resources available. The rather ambitious set of project objectives, which resulted from this process, is described in Section 1.3.

1.3 - PROJECT OBJECTIVES

At a high level, the project objective was; to produce a flood forecasting system that could provide Disaster Managers and other relevant authorities with advance warning of floods to allow them the lead-time necessary to put into action flood mitigation plans. This primary aim leaves a lot of room for interpretation and was sub-divided into a number of tasks required to meet the objective as well as the requirements described in Section 1.2.

Following is a list of the tasks that served as the basis for the project work program. The list was drawn up during the planning stages of the project and in some cases the tasks have been modified; such is the nature of research. These modifications will be discussed in the concluding chapter as well as other relevant places in the report.

Accessing Rainfall Data:

- *Ensure the reliability of the Durban Radar*. This required liasing with the SAWS (METSYS) to ensure that they could deliver a product of suitable quality. It was considered important that the radar be reliable and function correctly when severe rainfall events occur. Additional calibration of the radar could be achieved using data from rain gauges owned and managed by Umgeni Water, eThekwini municipality and DWAF.
- *Understand and write software to handle MDV format data*. Software to convert MDV data files to bitmap format (A convenient and widespread image format) has already been developed. The extraction of relevant rain rate information directly from the MDV format would reduce the online computational burden and was therefore preferred. Input from METSYS in terms of understanding the structure of the file format was required.
- *Obtain the MDV data stream directly in real-time*. METSYS already receives and archives the MDV data, from each of the radars, at their Bethlehem offices. The use of similar technology would enable the data stream to be received directly at the flood forecasting centre.
- *Generate catchment masks*. The rainfall data was to be selectively masked in order to determine the rainfall over each catchment area of concern. Such masks were to be generated from GIS data and converted to a bitmap (or other suitable) format. This information could be obtained from Umgeni Water's GIS coverages. The masks needed to be scaled to the same resolution as the rainfall field data.

Rainfall-runoff Model Fitting:

- *Gather relevant historical data sets*. The task here was to determine where relevant streamflow and rainfall data were available and how long the records are. This information was to be a combination of data from Umgeni Water and DWAF.
- *Implement Optimisation procedure*. This would require the use of sophisticated optimisation algorithms to determine the "best fit" parameter set for the chosen model to represent the catchment.
- *Address the issue of stability*. A scheme needed to be developed to ensure that only stable starting values could be selected for the Flood forecasting model. This should be guaranteed by starting only on an upturn of the flow hydrograph.

- *Selection of Parameter sets*. A number of plausible parameter sets for the Flood forecasting procedure were to be chosen and used for validation.

Forecasting rainfall fields and inclusion of the String of Beads Model:

- *Set up a forecasting version of the rainfall-runoff model*. This required the writing of software to accept the real-time data stream and convert it to rainfall inputs for the rainfall-runoff model. An online correction procedure also needed to be developed and included in the software implementation.
- *Set up a conditional forecasting implementation of SBM*. The "String of Beads" rainfall simulation model (SBM) developed under two successive WRC contracts is able to provide short term (a few hours ahead) conditional forecasts. Software was to be written to ensure that this forecasting facility could be used in real-time and updated as new data arrived.
- *Incorporate SBM in parallel forecast streams*. This task required the integration of the real-time flood forecasting rainfall-runoff model and SBM into a single unit (hereinafter called the Flood Forecasting System) with each running as a separate parallel stream. The conditional forecasts from the SBM model were to be used as input to the rainfall-runoff forecasting model to further improve its estimates of future flows.
- *User interface*. The core functions and console-based interface used in the testing phase would need to be replaced with a user-friendly front end. This task was thought to be beyond the scope of this project, but was envisioned as a future requirement.

Validation and testing:

- *Validation on "unseen" historical data (offline)*. It was necessary to ensure that the Flood Forecasting System (with its associated parameter set) provided useful flow estimates for Rainfall-Runoff sequences other than those used in the calibration procedure.
- *Validation and testing in real-time (online)*. Once the system had been put in place and properly tested in an offline mode, operational testing needed to be carried out in order to determine the reliability of the system. It was considered very important that flood warnings are believable and taken seriously by the disaster managers.

This chapter provides an overview of the various elements of the flood forecasting system and the rationale for each of the choices made. This is followed by a thorough description of each component of the research carried out and the techniques employed.

2.1 - STRUCTURE OF THE FLOOD FORECASTING SYSTEM

Figure 2.1.1 shows a schematic of the flood forecasting system devised to meet the requirements set out in Section 1.2. The arrows joining each element indicate the data flow within the system. The schematic highlights the importance of having observations of the inputs (rainfall) and outputs (streamflow) for the catchment being modelled. The first important element is the determination of the best estimate of the spatial rainfall field (BSR; Section 2.3) for input to the model. This estimate of rainfall is obtained from a combination of rain gauge and radar estimates, with the option to include satellite information based on the work done in the Spatial Interpolation and Mapping of Rainfall (SIMAR) programme (WRC, K5-1151/2/3). The BSR is used as direct input to the catchment model (current rainfall observation) as well as providing the basis for forecasting future rainfall fields using the rainfall-forecasting model (Section 2.4). The resulting short term forecasts of the expected evolution in space and time of the rainfall field provide further input to the catchment model to reduce the forecast uncertainty.

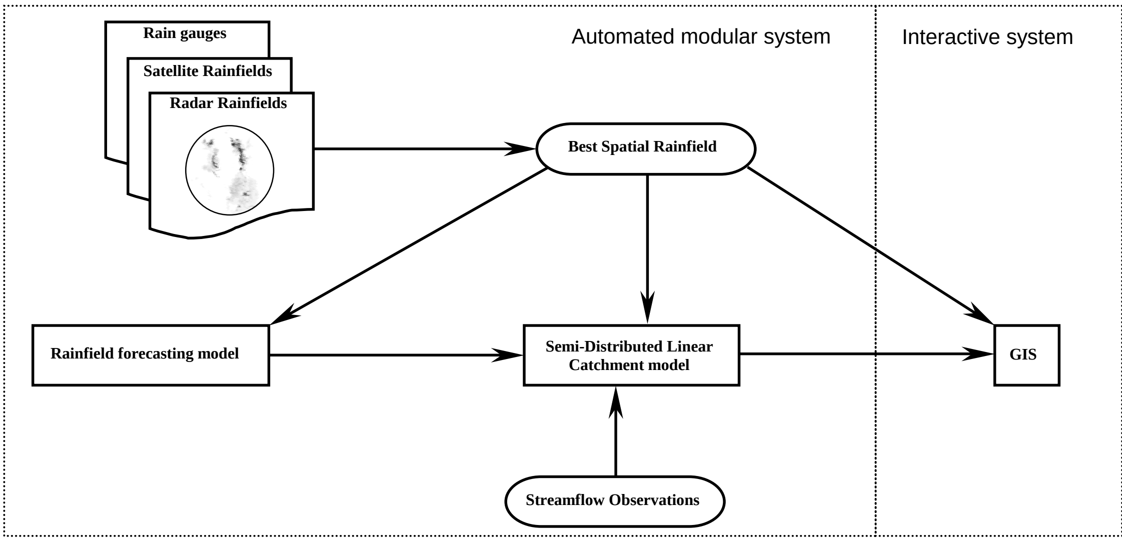


Figure 2.1.1 A Schematic of the proposed flood forecasting system

The semi-distributed linear catchment model proposed for this study makes use of streamflow observations and advanced Kalman filtering techniques to provide optimal forecasts of streamflow at the catchment outlet (Section 2.2). The processes described up to this point form a modular automated system requiring no user intervention. The user interacts with the system through the GIS interface. The current and forecast rainfall fields are processed into a format compatible with the Arcview GIS and may be viewed as overlays to the other themes. The forecast streamflows are displayed in terms of their corresponding floodlines in the GIS (Section 2.6).

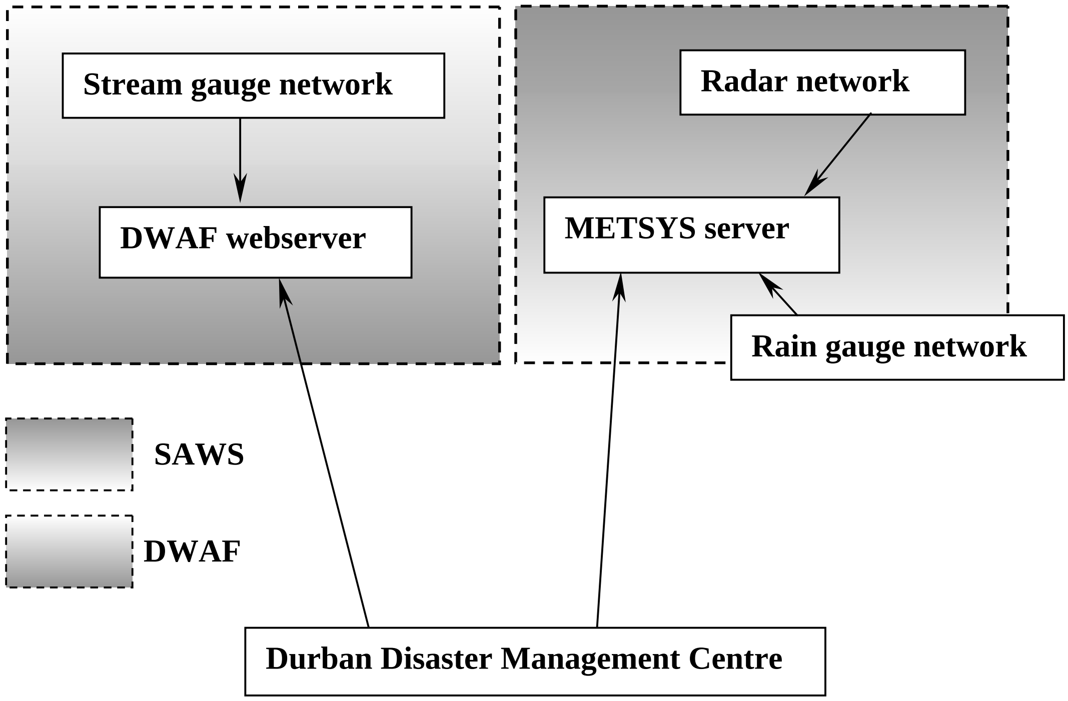


Figure 2.1.2 Data flow between organisations

A major requirement for a functional flood forecasting system is the availability of timely and accurate data. Figure 2.1.2 shows the data flows between some of the organisations involved in this study. Real-time streamflow data is collected at the Inanda and Shongweni dams by DWAF. The collection and transfer of data is handled by DWAF within their own data network and published on their web page five times daily. This data is automatically collected off the web page by software running at the eThekwini Municipality's Disaster Management centre.

In a similar way, the radar data collected by METSYS is processed into the form required before being stored on the METSYS server. This data is collected by a File Transfer Protocol (FTP) link from the Disaster Management centre. The data collection process is described in more detail in Section 2.6.

2.2 - CATCHMENT MODEL

The catchment model selected for use in this study is composed of an arrangement of three interconnected linear reservoirs, parameterised by the reservoir response constants and including a loss term for each reservoir; a complete description of the model is given by Pegram and Sinclair (2002). The model is used in a semi-distributed form where each portion of the catchment, WR90 quaternary catchments in this case, is treated in a lumped manner with a mean areal rainfall input as described in Section 2.5. For the purposes of making forecasts, the model is formulated according to its governing state-space equations rather than in the more efficient pseudo-ARMA form. The reasoning behind doing this was to take advantage of the Kalman and MISP (Todini, 1978) filters for online forecast updates. Updating of forecasts is an essential way to ensure that new information is incorporated as soon as it becomes available.

Since the forecasting system has been designed in a modular manner, the catchment model could be replaced with a more complex (physically based or conceptual) model that is considered to be representative of the catchment. The rainfall forecasts will become a more important aspect of the system in this case (Lardet and Obled, 1994). However, this may mean that the recursive updating` schemes proposed here cannot be used if the formulation of the replacement model can't be adapted appropriately. This is often the case for complex "off the shelf" modelling solutions. It follows that a decision should be made on an individual basis for each catchment that the system is applied to.

The advantage of recursive estimation techniques (such as the Kalman filter) is that they only require the measurements from the immediate past in order to provide the optimal (least squares) estimates; this makes them ideally suited to online computational situations. A further important benefit is the routine computation of the error covariance matrix, which can provide an estimate of the forecast confidence at each time step.

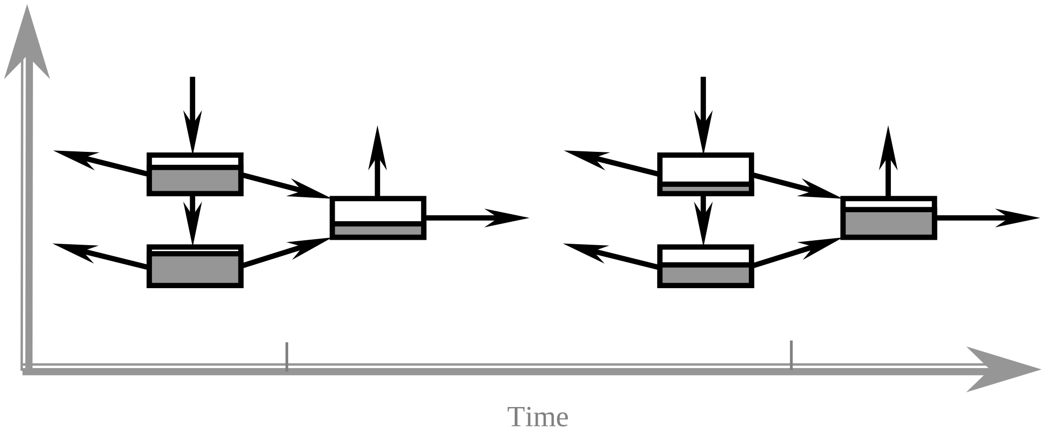


Figure 2.2.1 Model state updates (note the different storage levels at the different epochs).

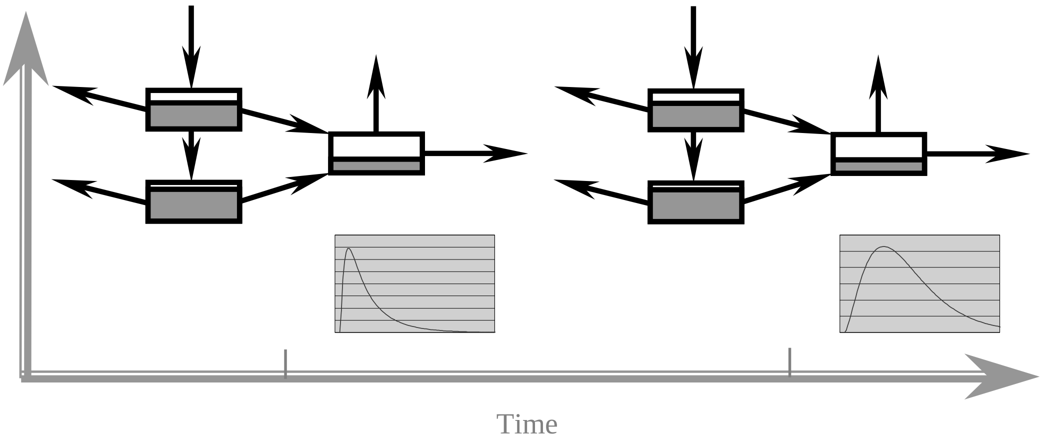


Figure 2.2.2 Model parameter updates (note the same storages, but differing responses).

Figures 2.2.1 and 2.2.2 indicate the difference between state updates and parameter updates, in the context of the catchment model used in this study. State updates refer to a change of storages within the model's reservoirs, while parameter updates change the way the model will respond to a given set of storages to reflect the changes in the catchments expected response with time. The Kalman filter (Kalman, 1960) updates the model states while the MISP algorithm (Todini, 1978) is a modified version the filter, which provides both state and parameter updates at each time-step.

2.2.1 - FORECAST UPDATES USING THE KALMAN FILTER

The Kalman filter is a recursive state estimation scheme. It produces optimal estimates (in a least squares sense) of a system's state vector x_t, where the system may be defined by the following linear stochastic difference equation

$$x_t = Ax_{t-1} + Bu_t + w_{t-1}$$

where A is the State transition matrix, B is the system input conversion matrix, u_t is the exogenous input vector and w_{t-1} is the system noise, which is assumed to be normally distributed white noise with mean $\overline{w}$ and covariance matrix Q.

The system measurement equation is given by

$$y_t = Hx_t + v_t \qquad \textbf{(2.2.1)}$$

Where y_t is the system measurement vector, H is the state to measurement transfer matrix and v_t is the measurement noise which is assumed to be normally distributed white noise with mean $\overline{v}$ and covariance matrix R.

The derivation of the filter equations can be found in various texts (e.g. Gelb, 1974; Young, 1984) and is omitted here. The filter equations fall into two categories: prediction and correction equations. The prediction equations provide *a priori* (forecast) estimates of the system states and state error covariance matrix; the correction equations compute the *a posteriori* estimates of the system states. The estimated measurements are given by equation 2.2.1. The matrices A, B, Q, R & H, the measurements y_t and the input vector u_t are all assumed to be known until time t; the unknowns are the states $\hat{x}_t$ and the noise terms w_t and v_t.

The filter is applied recursively with initial estimates of the system states and associated errors being made to start the filter. The prediction equations are used to forecast to the next time step and the correction equations applied once a measurement becomes available. This is illustrated in Figure 2.2.3, along with the filter equations. In the figure $\hat{x}_{t+1/t}$ is the *a priori* estimate of the system states, $\hat{x}_{t/t}$ the *a posteriori* estimate of the system states, $P_{t+1/t}$ the *a priori* estimate of the system error covariance matrix and $P_{t/t}$ the *a posteriori* estimate of the error covariance matrix. K_t is the gain of the filter.

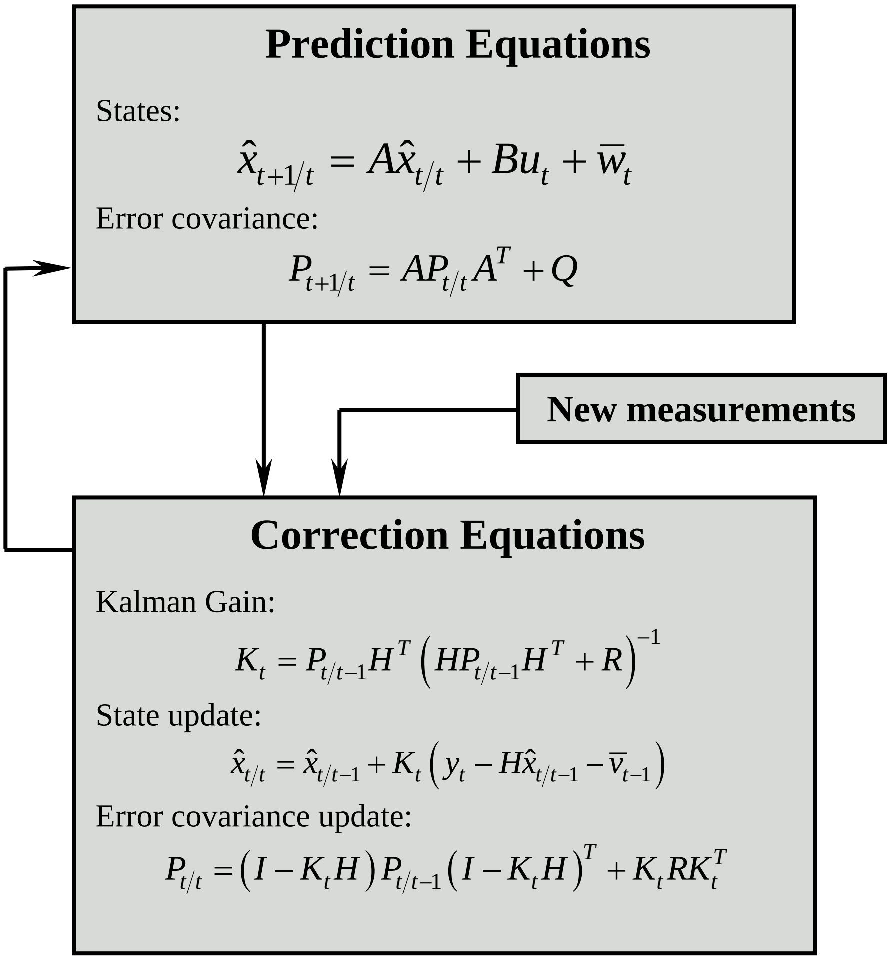


Figure 2.2.3 The Kalman filter cycle

2.2.2 – THE MISP ALGORITHM

The MISP algorithm due to Todini (1978) attempts to solve both the parameter estimation problem and the state estimation at each time step. In order to achieve this goal the algorithm includes a Kalman filter on the model's parameters, allowing the optimum model parameter set to be computed concurrently with the state updates. The equations and description of terms for MISP are given in Appendix A.

2.2.3 - RESULTS OF THE UPDATING ALGORITHMS IMPLEMENTED

A software implementation of the MISP and Kalman filters was developed and tested for use in a forecasting version of the catchment model. Figures 2.2.4 and 2.2.5 show example forecasts made with lead-times of up to twelve hours. The forecasts were made for flood events on the Liebenbergsvlei catchment in the Free State province (This set of data is used here for descriptive purposes because good synchronised rainfall-runoff data are hard to come by in the Umgeni for shorter than daily timesteps). It is clear from these figures that the accuracy of the forecasts reduces with lead-time. This is an inescapable truth as the confidence with which forecasts can be made reduces with lead-time even for a perfect catchment model, since the rainfall input cannot be known in advance. It is pleasing to note that the general character of the events is retained even by the twelve hour ahead forecasts, as this indicates an appropriate model response to the observed rainfall.

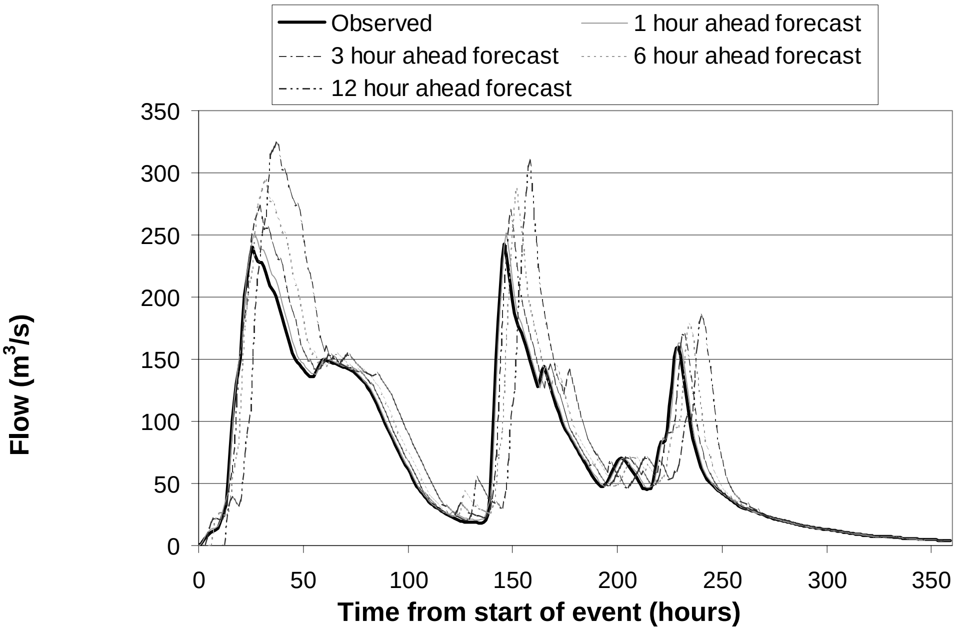


Figure 2.2.4 Forecasts made using the Linear reservoir model and the Kalman filter (December 1995, Liebenbergsvlei)

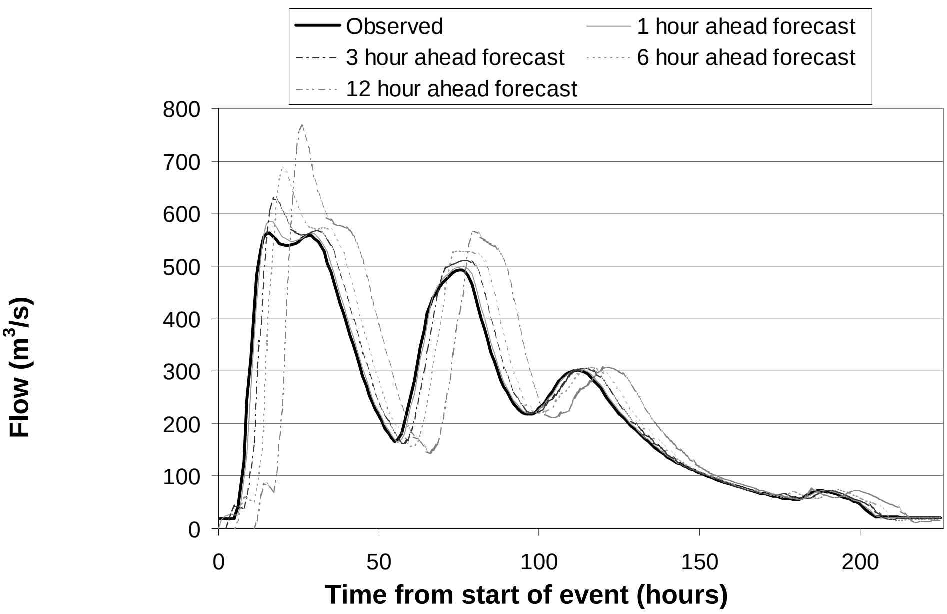


Figure 2.2.5 Forecasts made using the Linear reservoir model and the Kalman filter (February 1996, Liebenbergsvlei)

Figures 2.2.6 and 2.2.7 give a comparison of the square root of the mean sum of squared forecast errors (R.M.S.S.E) for forecasts made using the linear reservoir model with Kalman filtering as well as for persistence forecasts. A persistence forecast is one in which the current value of streamflow is used as the forecast value for all lead-times. This is equivalent to assuming that no change in streamflow is expected as time goes on. Persistence forecasts take no cognisance of the streamflow history and Figures 2.2.6 and 2.2.7 show clearly that the Kalman filtered model forecasts outperform the persistence forecasts, due to their incorporation of the characteristics of the recent flows.

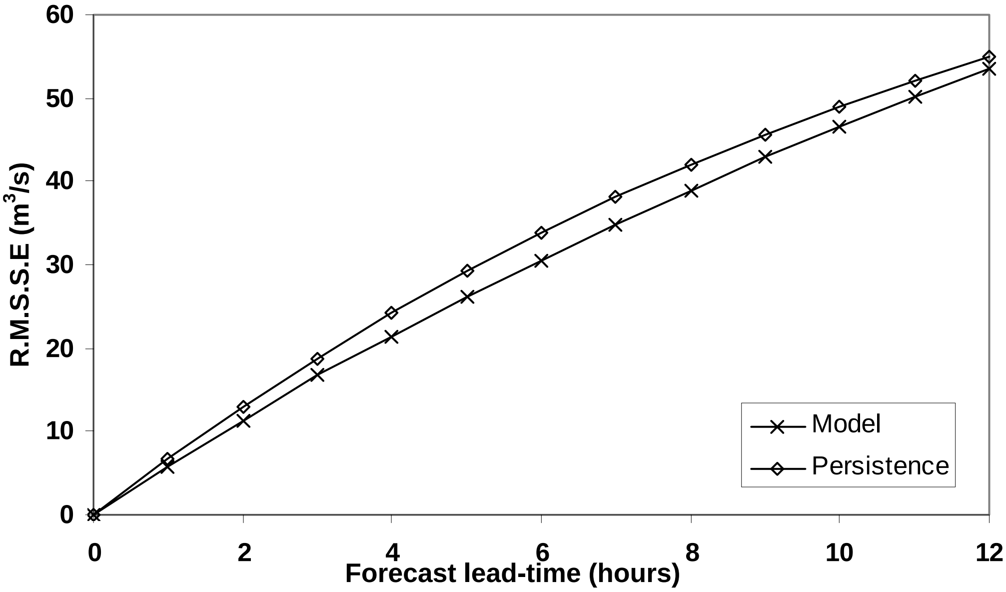


Figure 2.2.6 Comparison of the root mean sum of squared forecast errors (December 1995, Liebenbergsvlei)

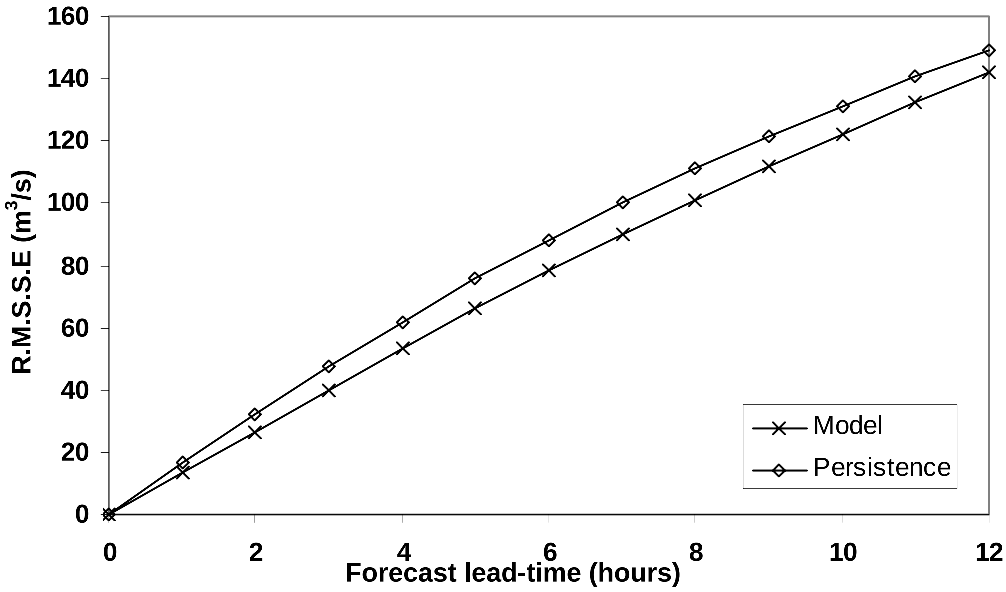


Figure 2.2.7 Comparison of the root mean sum of squared forecast errors (February 1996, Liebenbergsvlei)

Figure 2.2.8 gives a comparison between observed and forecast streamflows for the 1995 event of Figure 2.2.4. The left hand plots show the comparison for forecasts made using the linear reservoir model and Kalman filter while the right hand side plots show the comparisons for persistence forecasts – note the larger R^2 values in the plots in the left hand column. Figure 2.2.9 shows the same comparisons for the 1996 event of Figure 2.2.5. These figures provide an alternative way of showing the model forecasts to be an improvement over the persistence forecasts.

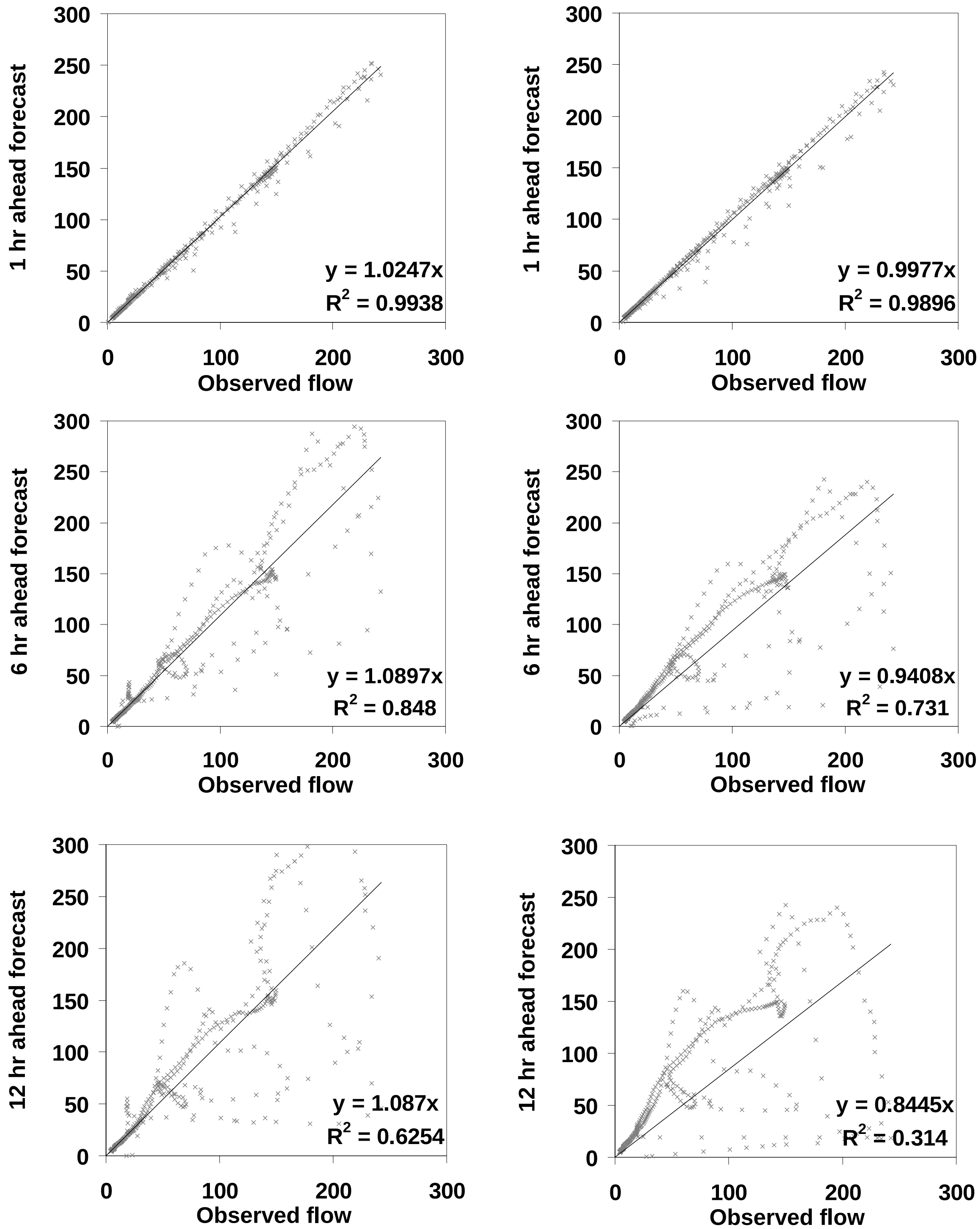


Figure 2.2.8 Observed and forecast scatter-plots (December 1995, Liebenbergsvlei)

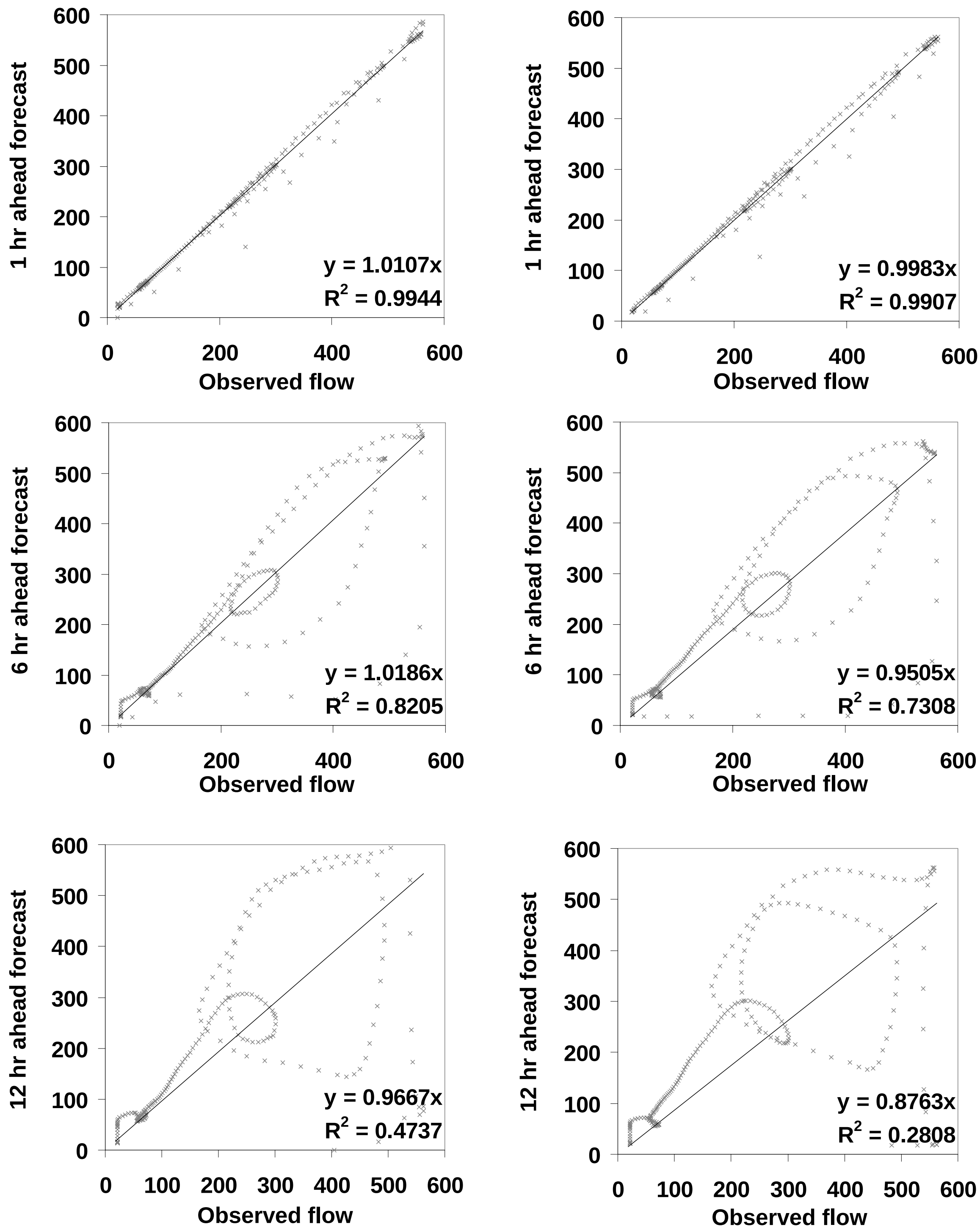


Figure 2.2.9 Scatter plots of observed and forecast flows (February 1996, Liebenbergsvlei)

Attempts to update the parameters in conjunction with the model states, using the MISP algorithm, did not produce satisfactory results. The parameters were found to have too many degrees of freedom to allow a satisfactory parameter set to be chosen for each time-step. The forecasts are therefore produced using only the Kalman filter to update model states at each time-step based on an initial estimate of the parameters. This aspect of forecasting should be given more attention in the future.

2.3 - RADAR AND RAIN GAUGE MERGING

Rain gauges are, traditionally, the tool used by hydrologists to provide information on the rainfall input to a catchment for the purposes of flood modelling. Although rain gauges may be subject to underestimation of heavy rainfall (Wilson and Brandes, 1979) and other errors, they can be considered to give relatively unbiased and reasonably accurate measurements of point rainfall and provide (importantly) a direct measurement of rainfall depth. The major disadvantage of these point measurements is the lack of detailed spatial detail information provided by the rain gauges (unless they are in a very dense network).

It is in the description of spatial detail where the weather radar provides a significant advantage over traditional techniques of interpolating between point estimates to provide an estimated picture of the true rainfall field in space and time. The weather radar provides an indirect measurement by relating the measured reflectivity (electromagnetic energy reflected from raindrops in the atmosphere) to rainfall rate. The reflectivity measurements are spatially averaged (over the radar bins) and converted from their polar co-ordinate form into a regular Cartesian grid format (Mittermaier and Terblanche, 1997; Clothier and Pegram, 2002). The rainfall estimated over each grid square is, in general, biased and subject to several forms of error which necessitate the use of correction procedures (Grecu and Krajewski, 2000).

Rain gauge information is commonly used in correction procedures to calibrate the parameters used in the radar reflectivity to rainfall rate conversion equation (Hannesen, 2002a). Due to the highly variable nature of rainfall in space and time and the different sampling characteristics of the two sensors, such comparisons can only be sensibly made for accumulations, over durations in the order of months to years. In the sections which follow, the process called “Merging” refers to the synthesis of rainfall information provided by rain gauges and radar into a single estimate of the true (and unknown) rainfall field at shorter timescales; in the range: minutes to hours. Such merging techniques allow the flood hydrologist to take advantage of the strengths of each instrument’s estimate of rainfall while taking cognisance of and minimizing the effect of its weaknesses. As a result the best possible estimate of the spatial rainfall field is obtained to reduce the uncertainty inherent in forecasting. The sections that follow provide a quantitative comparison between two recently proposed merging techniques by means of a numerical experiment, carried out to decide on the most appropriate technique to use in the flood forecasting system. The techniques investigated were the Bayesian merging technique proposed by Todini (2001) and the Conditional merging technique of Ehret (2002), based on a suggestion by Pegram (2002).

2.3.1 - DESCRIPTION OF THE BAYESIAN MERGING TECHNIQUE

The Bayesian merging technique (Todini, 2001) is a means to provide an optimal combination of radar and rain gauge rainfall estimates. The algorithm requires Kriging the rain gauge measurements of rainfall onto a grid with the same spatial resolution as that of the radar rainfall estimates. Following the Kriging step a single recursion of the Kalman filter (Kalman, 1960) produces the optimal (in a Bayesian sense) estimate of the “true” unknown rainfall field. This estimate is only optimal insofar as the “true” rainfall field can be adequately modeled as a spatially correlated field of Gaussian noise.

The sequence of computations can be described by referring to equations 2.3.1 to 2.3.8

- The first step is to Krige the point rain gauge information onto the radar grid to obtain the unbiased and minimum variance estimate of the unknown true rainfall field, given the information content available from the rain gauges (Equations 2.3.1, 2.3.2).
- The *a priori* estimates of the true field y_t' and its error covariance P_t' are given by equations 2.3.3 and 2.3.4, where the radar's expected bias and error structure are derived from historical observations, of the errors between the Kriged gauge and radar fields.
- The innovations vector ν is computed from equation 2.3.5 assuming the Kriged gauge field to be the available observation of the state vector.
- The Kalman gain matrix K_t is computed from equation 2.3.6.
- The *a posteriori* estimate of the field y_t'' and its error covariance P_t'' are computed from equations 2.3.7 and 2.3.8 using the Kalman gain matrix to modify the *a priori* estimates as shown.

Kriging

$$y_t^G = \Lambda x_t^G \qquad \textbf{(2.3.1)}$$

$$\left[\Lambda \;\vdots\; \mu\right] = \left[\Gamma_{lb} \;\vdots\; u\right]\begin{bmatrix}\Gamma & \vdots & u \\ \cdots & & \cdots \\ u^T & \vdots & 0\end{bmatrix}^{-1} \qquad \textbf{(2.3.2)}$$

Kalman Filter

$$y_t' = y_t^R - \mu_{\varepsilon_t^R} \qquad \textbf{(2.3.3)}$$

$$P_t' = V_{\varepsilon_t^R} = V_{\varepsilon_t} + \left\{\left[\Lambda \;\vdots\; \mu\right]\begin{bmatrix}\Gamma & \vdots & u \\ \cdots & & \cdots \\ u^T & \vdots & 0\end{bmatrix}\begin{bmatrix}\Lambda^T \\ \cdots \\ \mu^T\end{bmatrix} - \Gamma_{ll}\right\} \qquad \textbf{(2.3.4)}$$

$$\nu_t = y_t^G - y_t' \qquad \textbf{(2.3.5)}$$

$$K_t = V_{\varepsilon_t^R} V_{\varepsilon_t}^{-1} = P_t' V_{\varepsilon_t}^{-1} \qquad \textbf{(2.3.6)}$$

$$y_t'' = y_t' + K_t \nu_t \qquad \textbf{(2.3.7)}$$

$$P_t'' = P_t' - K_t P_t' \qquad \textbf{(2.3.8)}$$

where

y_t^R	is the vector of radar rainfall estimates on each pixel at time t.
x_t^G	is the vector of rain gauge rainfall estimates at time t.
y_t^G	is the vector of Kriged gauge rainfall estimates on each grid square at time t.
Λ	is the matrix of Kriging weights.
μ	is a vector of Lagrange multipliers.
Γ	is the covariance matrix among the rain gauge locations.
Γ_{lb}	is the covariance matrix between the radar grid squares and rain gauge locations.
u	is a unit vector.
y_t'	is the *a priori* estimate of the Kalman filter state vector (the "true" rainfall field).

$\mu_{\varepsilon_t^R}$ is the vector of mean radar estimation errors on each grid square (from past observations).

P_t' is the *a priori* estimate of the Kalman filter state error covariance matrix.

$V_{\varepsilon_t^R}$ is the covariance matrix of radar estimation errors

V_{ε_t} is the covariance of the measurement difference time series ($\varepsilon_t = y_t^R - y_t^G$).

Γ_{ll} is the covariance matrix among the radar grid squares.

ν_t is the vector of Kalman filter innovations.

K_t is the Gain matrix of the Kalman filter.

y_t'' is the *a posteriori* estimate of the filter state (the "true" rainfall field).

P_t'' is the *a posteriori* estimate of the state error covariance matrix.

Because the algorithm is computationally intensive it is useful to list the number of operations required for each step. Denoting the number of radar pixels (the 1km x 1km radar grid squares) by *m* and the number of gauge measurements by *n*. The operation count is as follows, with typical numbers of floating point operations (FLOPS) given in parentheses for *m* = 40 000 and *n* = 50:

(2.3.1) [m, n] matrix by [n] vector multiplication (2×10^6).

(2.3.2) [n+1, n+1] matrix inverse *and* [m, n+1] by [n+1, n+1] matrix multiplication (625 000, 2×10^6).

(2.3.3) [m] vector subtraction *or* Scalar subtraction from an [m] vector (40 000).

(2.3.4) [m, n+1] by [n+1, n+1] *and* [m, n+1] by [n+1, m] matrix multiplications *and* two [m, m] matrix subtractions (10^8, 4×10^{12}).

(2.3.5) [m] vector subtraction (40 000).

(2.3.6) [m, m] matrix inverse *and* [m, m] by [m, m] matrix multiplication (64×10^{12}, 256×10^{16}).

(2.3.7) [m, m] matrix by [m] vector multiplication *and* [m] vector addition (64×10^{12}).

(2.3.8) [m, m] by [m, m] matrix multiplication *and* [m, m] matrix subtraction (256×10^{16}).

The magnitude of matrix arithmetic obviously increases with the size of the radar field that is to be conditioned, making the method computationally infeasible for very large fields. The obvious workaround is to use moving window methods and split the field into a number of smaller areas. An extreme case of this thinking is to treat each pixel in the field separately, as was done in the large scale (128x128) experiments reported on in Section 2.3.5, this reduces the size of m to one making huge savings in the computational load.

The disadvantage of the moving window approach described above is the loss of information from neighbouring pixels. Examination of equation 2.3.7 shows that the estimate of the "true" field at each pixel is a combination of the *a priori* estimate at that pixel and a linear sum of the innovations at each pixel in the field. The combination of the innovations at each pixel is determined by the Gain matrix computed from equation 2.3.6. If we can safely assume that the correlation structure is such that the spatial dependence dies off quickly then we can use the more computationally efficient construct. A 7x7 lattice was used to give a determination of the error to be expected from using the method suggested here (pixel at a time) rather than the full method as described by Todini. The results show that there appears to be no substantive difference between the two methods. It is further postulated that as the fields become larger the computational advantages will outweigh the slight loss of accuracy, particularly once the dimensions of the field become much larger than the expected spatial correlation lengths associated with common rainfall field structures.

2.3.2 - VERIFICATION OF THE BAYESIAN MERGING IMPLEMENTATION

Todini (2002) provides a numerical experiment to demonstrate the potential of the proposed algorithm. This section describes the experiment conducted to replicate Todini's experiment and therefore verify that the coding of the relatively complex algorithm was accomplished correctly. This exploratory computation demonstrates the results achieved in terms of bias and variance reduction. The first step was to generate 2000 independent standard normally distributed, N(0, 1), realizations of a random field on a 7x7 lattice and sample the lattice at 9 gauge measurement points (Figure 2.3.1). The lattice points were labelled row-wise from 1 to 49, starting at the top left hand corner.

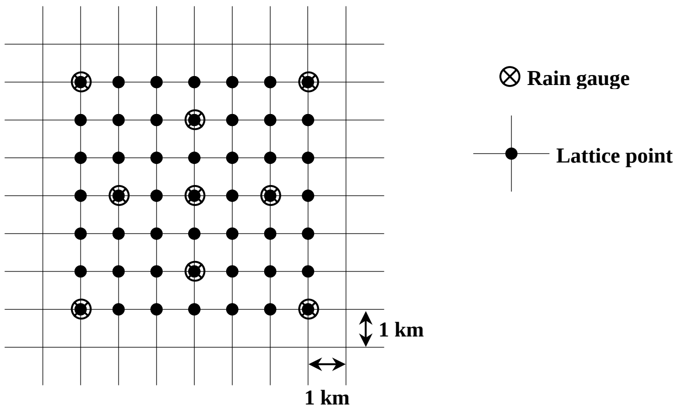


Figure 2.3.1 A Schematic of the pixel lattice and gauge measurement points

Next, a spatial correlation structure was imposed on the noise field in the following manner.

Compute the matrix B where BB^T is defined by

$$BB^T = \Gamma_{ll}$$

Γ_{ll} is the covariance matrix amongst the lattice points and is computed from equation 2.3.9 using a Gaussian variogram function.

$$v(h) = \sigma^2 - \left[p + \omega \left(1 - e^{-h^2/a} \right) \right] \qquad \textbf{(2.3.9)}$$

where σ^2 is the field variance, p is the nugget, ω the sill, a the range and h the distance between points.

B is computed by singular value decomposition (SVD). SVD states that any matrix A may be decomposed as follows

$$A = UWV^T$$

if we set $A = BB^T$, then

$$B = UW^{1/2}V^T$$

we can easily verify that $BB^T = A$ as we expect

$$BB^T = UW^{1/2}V^T VW^{1/2}U^T = UWV^T = A$$

since $U = V$ and $V^T = V^{-1}$ if A is a square symmetric matrix (Press *et al*, 1992). The "true" rainfall field y_l is computed from the 49-element vector of N(0, 1) random noise δ_l by premultipling with B

$$y_l = B\delta_l$$

The gauge measurements are considered unaffected by measurement errors, but a noise field (with it's own covariance structure) is generated on the lattice and added to the observations to produce the simulated radar measurements. The noise field is generated in the same way as the "observed" field, but with different variogram parameters (as shown in Table 2.3.1).

	Mean (μ)	Variance (σ^2)	Nugget (p)	Sill (ω)	Range (a)
Rainfall field parameters	0	10000	0	10000	10^7
Radar noise field parameters	40	3000	0	3000	10^6

Table 2.3.1 Variogram parameters

The statistics of the "true" rainfall field and the noisy radar measurement of the true field are shown in Figures 2.3.2 and 2.3.3.

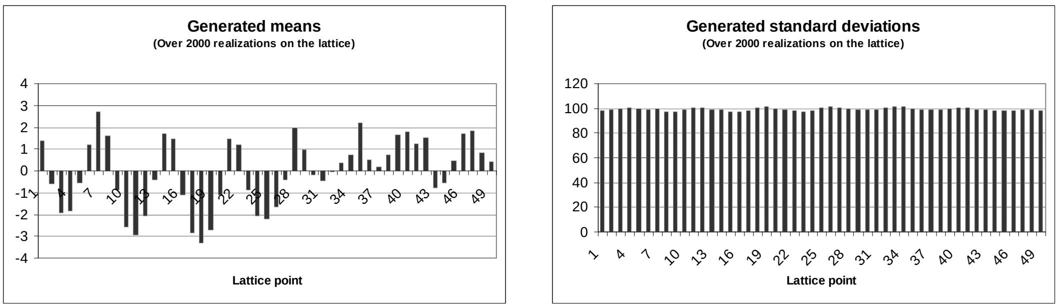


Figure 2.3.2 Statistics of the "true" field generated on the lattice points

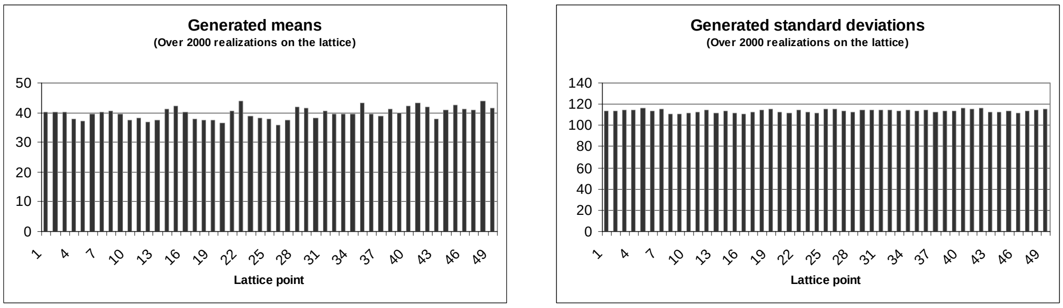


Figure 2.3.3 Statistics of the noise affected radar observations of the "true" field

Figures 2.3.4 to 2.3.6 show a selection of the different fields for a single realization to allow for an intuitive comparison.

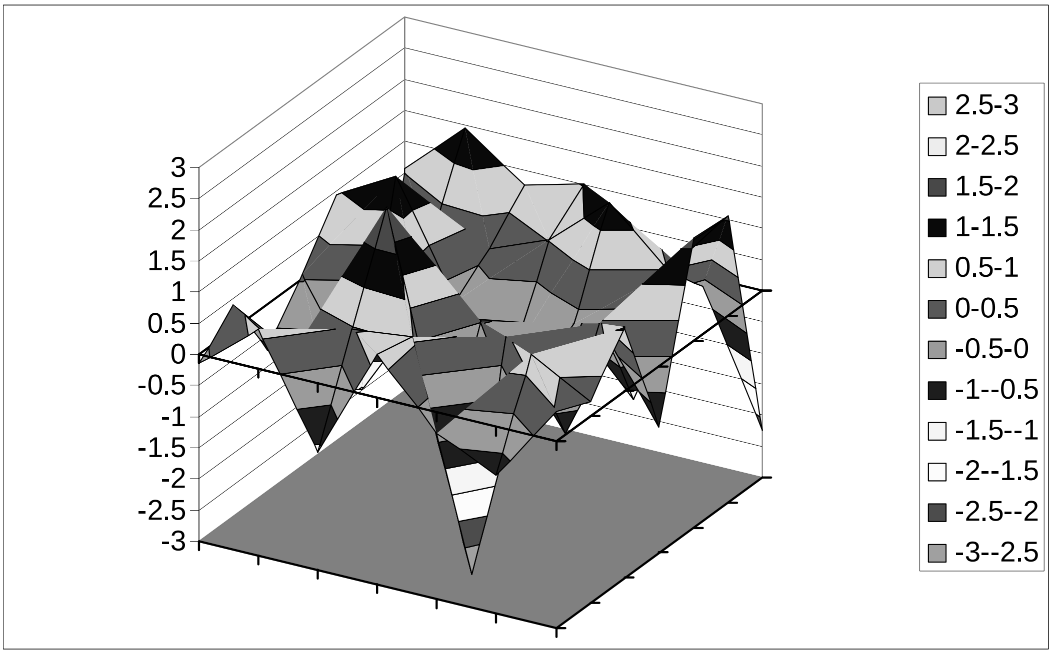


Figure 2.3.4 A single realization of a spatially uncorrelated Gaussian random field

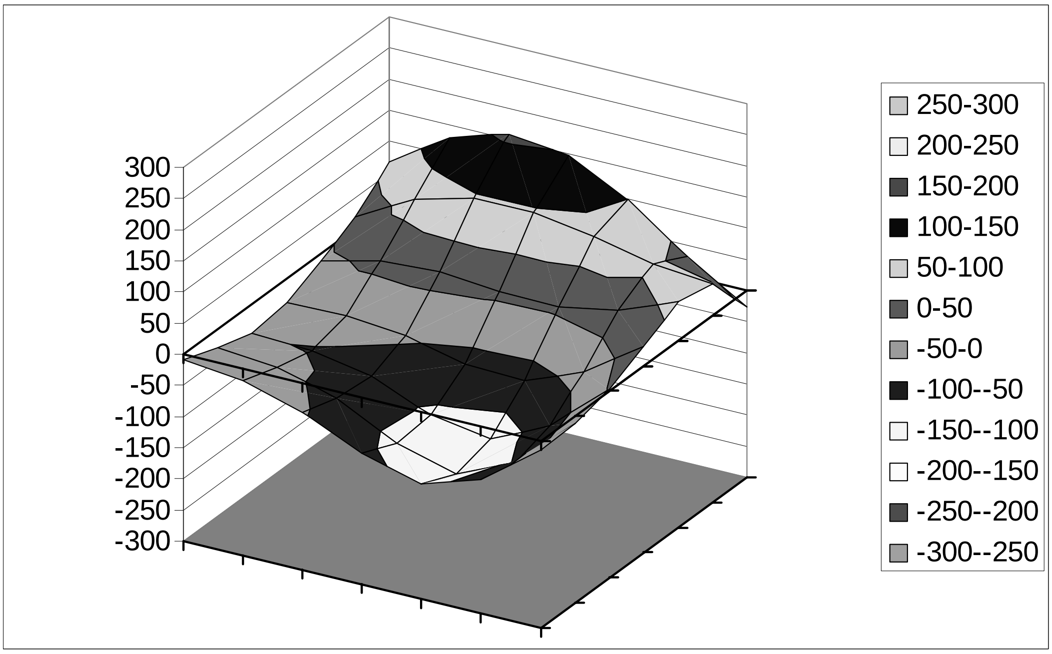


Figure 2.3.5 A spatially correlated random field (the “true” rainfall field)

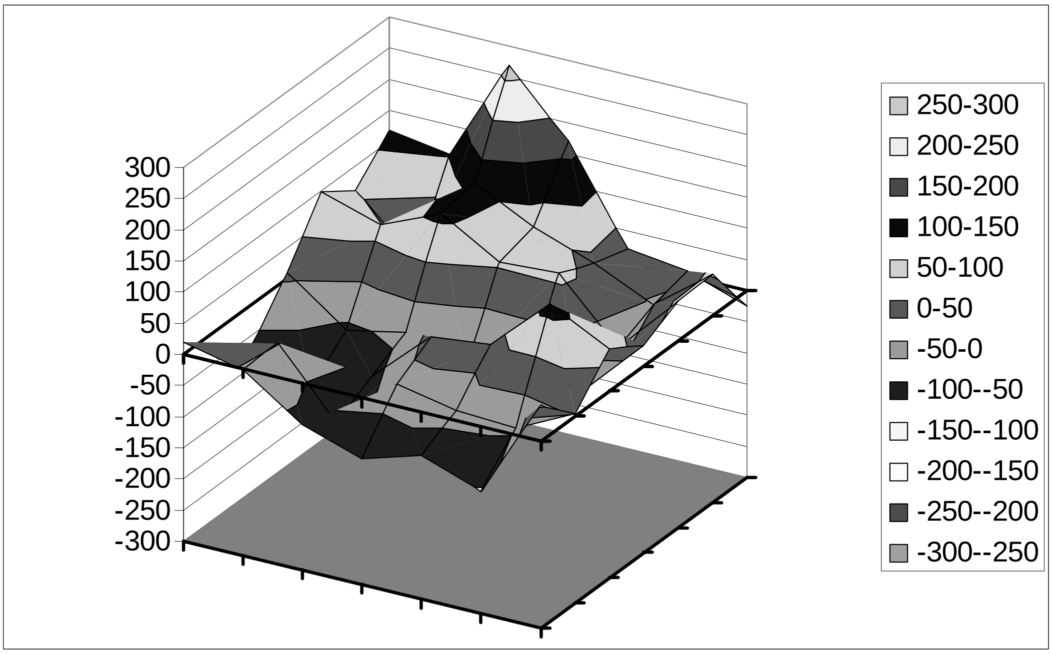


Figure 2.3.6 Noise affected radar observation of the "true" field

Before applying the full Bayesian merging technique, a check was made on the Kalman filtering routine. In this case the rainfall field y_t^G (obtained by Kriging the gauge measurements onto the lattice points) was replaced with the true field. In this way the efficiency of the Kalman filter in retrieving the original field was tested. The example fields above represent a single realization on the lattice and the results of applying the Kalman filter. The prior field is obtained by subtracting $\mu_{\varepsilon_t^R}$, the mean of the radar noise, from the radar field (Equation 2.3.3). It is clear that the posterior estimate of the true field is identical to the original as it should be.

Figure 2.3.7 shows a plan view of the true field and a noisy radar estimate of the true field for a single realization on the lattice. The figure also shows the *prior* and *posterior* estimates of the true field from the Kalman filter, where the prior field y_t' is obtained from equation 2.3.3. The comparisons in Figure 2.3.7 indicate that a large part of the bias is removed by subtraction of the radar error mean, to produce the prior estimate of the field. The application of the Kalman filter removes all traces of bias and also reduces the standard deviation between the posterior estimate and the true field to zero, as it should do in this case.

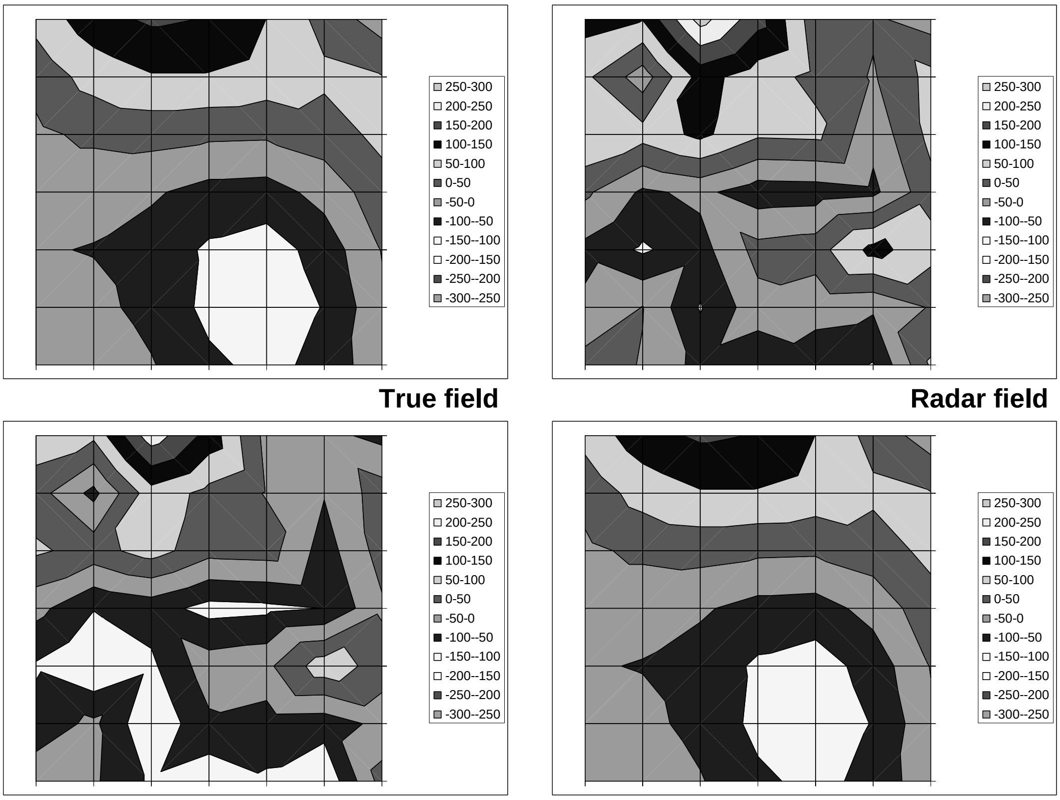


Figure 2.3.7 Verification of code: Example fields representing a single time realization, with the Kriged gauge field replaced by the true field

A further check was carried out to verify that the Kriging portion of the code was operating correctly. In this case the set of gauge measurements x_t^G was increased to be the 49 true field measurements on the 7x7 lattice. The Kriging procedure was employed to Krige these values back onto the lattice points, evidently the original field should be recovered exactly, as the matrix of Kriging weights Λ would become the identity matrix. This correspondence is shown in Figure 2.3.8.

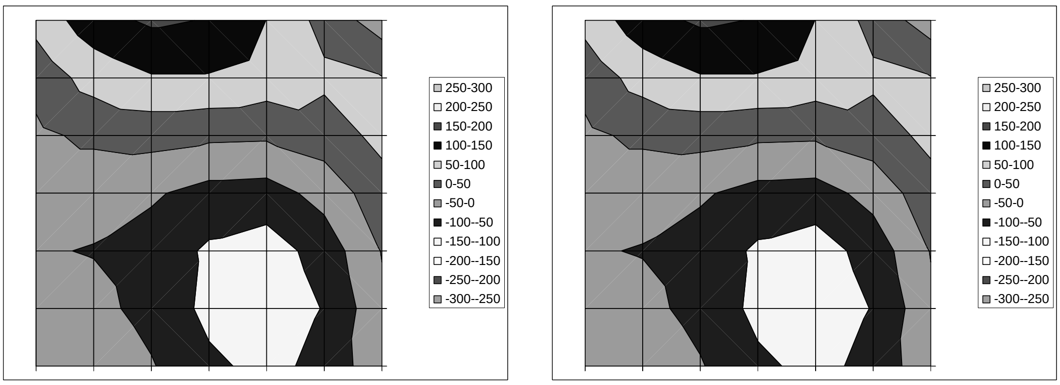


Figure 2.3.8 Verification of code: Example fields representing a single time realization, x_t^G was set as the “true” field.

The mean and standard deviation for the Kriged residuals (the difference between the "true" and Kriged fields) were also computed for 2000 applications of the Kriging routine where now the error and bias was reintroduced into the radar fields (Figure 2.3.9). The negligible errors of estimation can be attributed to floating point precision errors in the computation.

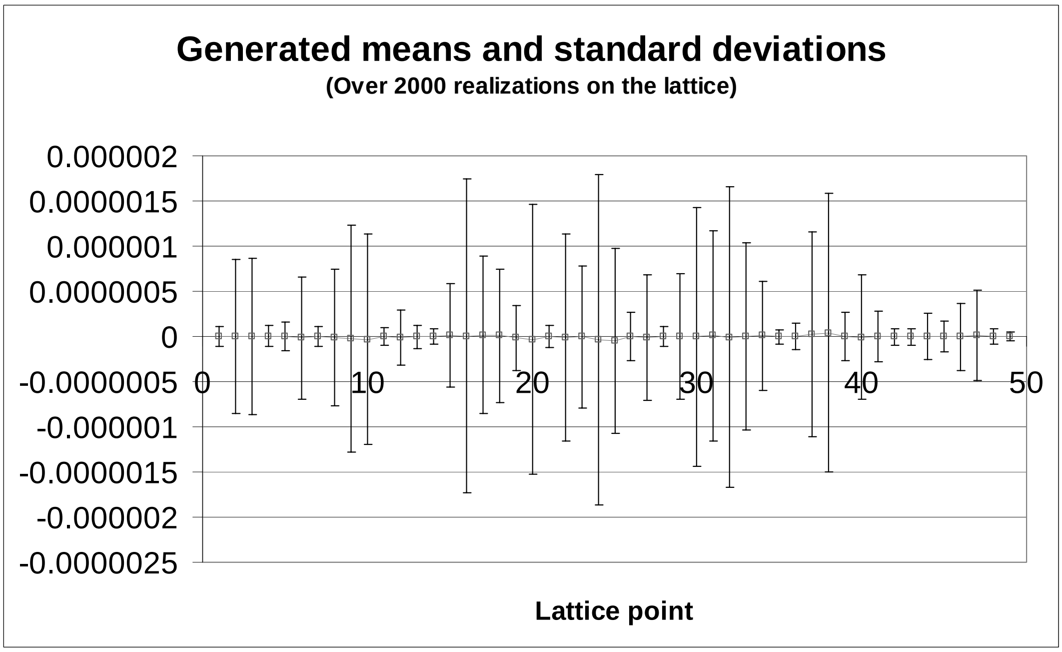


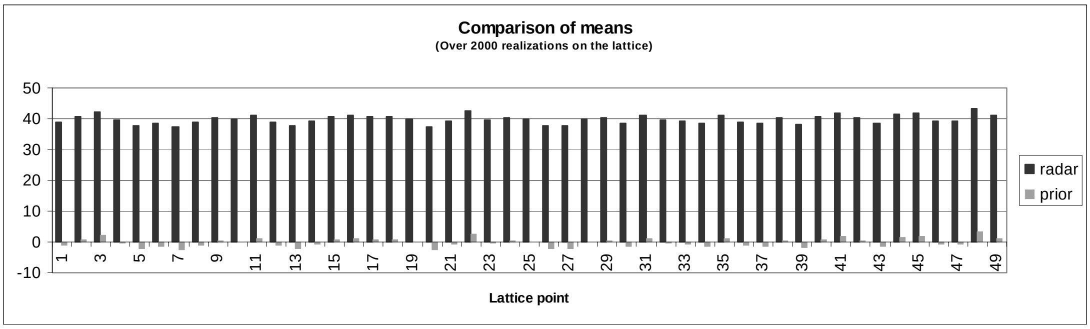
Figure 2.3.9 Computed statistics for the Kriged residuals

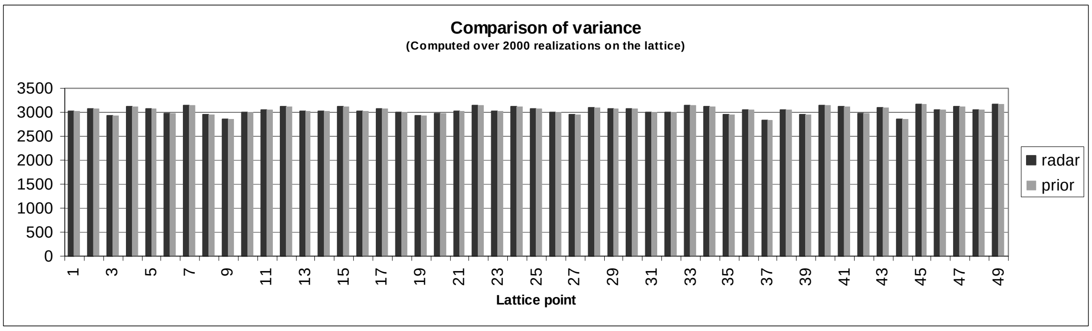
Having thoroughly checked the coding of the algorithm, the Bayesian merging technique was then implemented with the mean and covariance structure of the residuals measured *a priori*. The sequence of residuals between the radar estimate of the true field y_t^R and the Kriged field estimate y_t^G was examined and the mean μ_{ε_t} and covariance V_{ε_t} computed from these residuals. The Kalman filter equations were then applied with

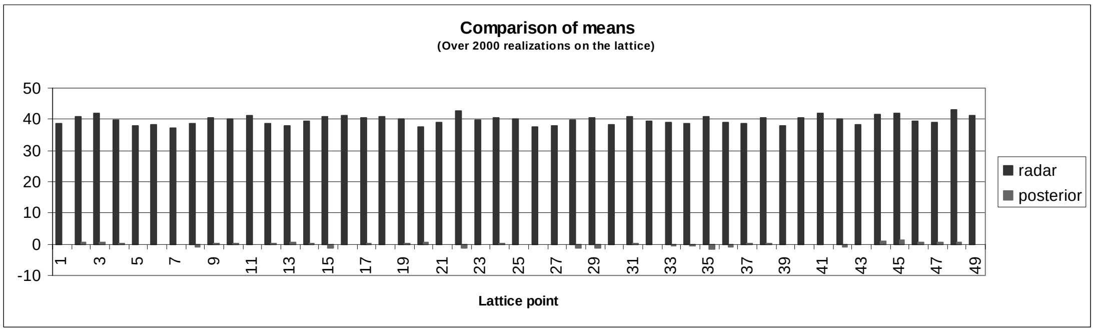
$$\mu_{\varepsilon_t^R} = \mu_{\varepsilon_t}$$

since the Kriged estimate y_t^G is assumed unbiased. The known covariance structure was used for $V_{\varepsilon_t^R}$. The results are summarized in Figures 2.3.10 to 2.3.12. It is clear that the mean errors become close to zero and that their variance shows a large reduction in magnitude.

These figures qualitatively match (the random number sequences are different) those of Todini (2001) very well, verifying and corroborating the technique.







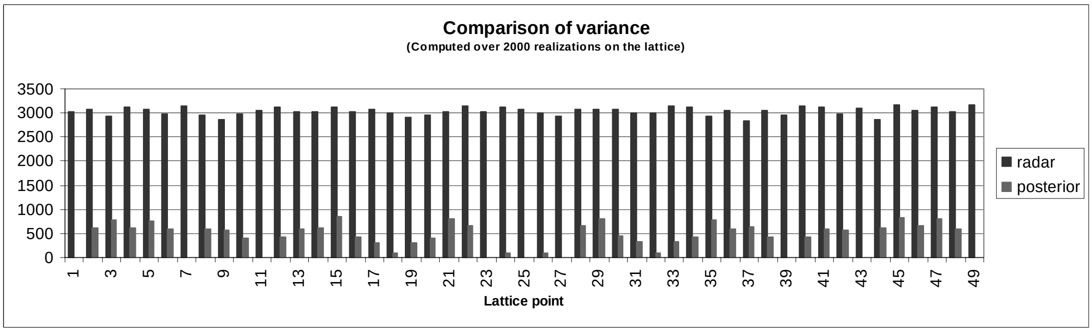


Figure 2.3.10 Comparison of statistics of residuals

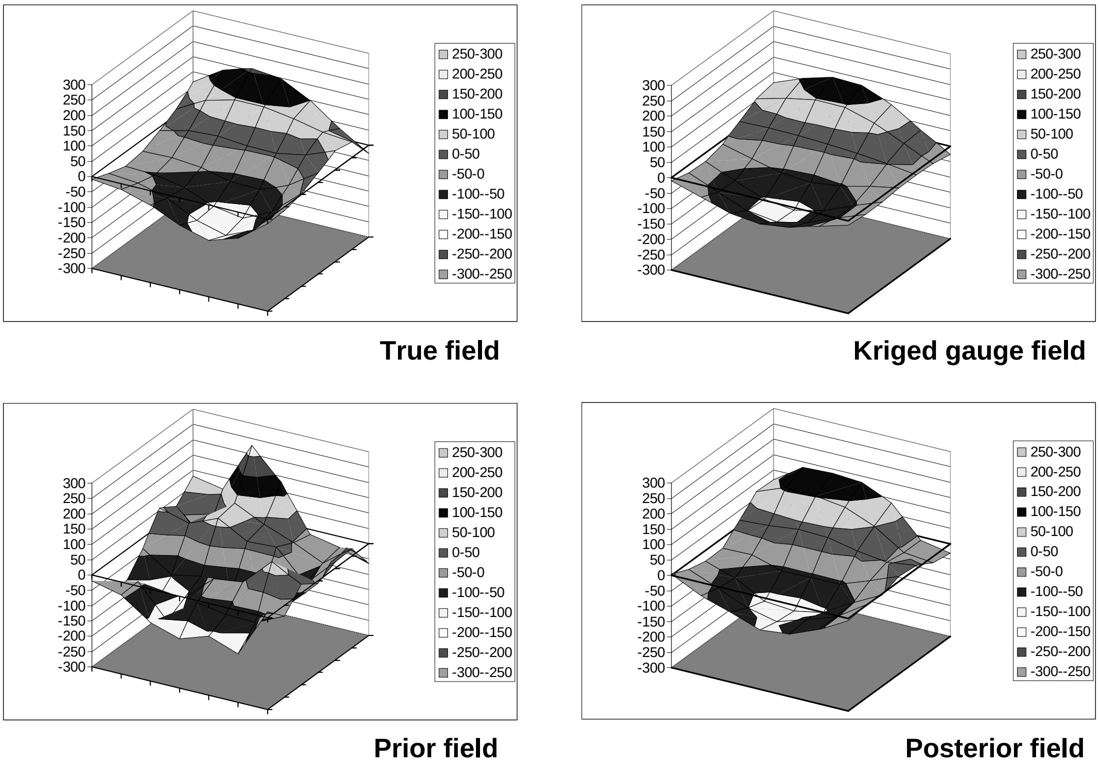


Figure 2.3.11 3D view of fields representing a single time realization

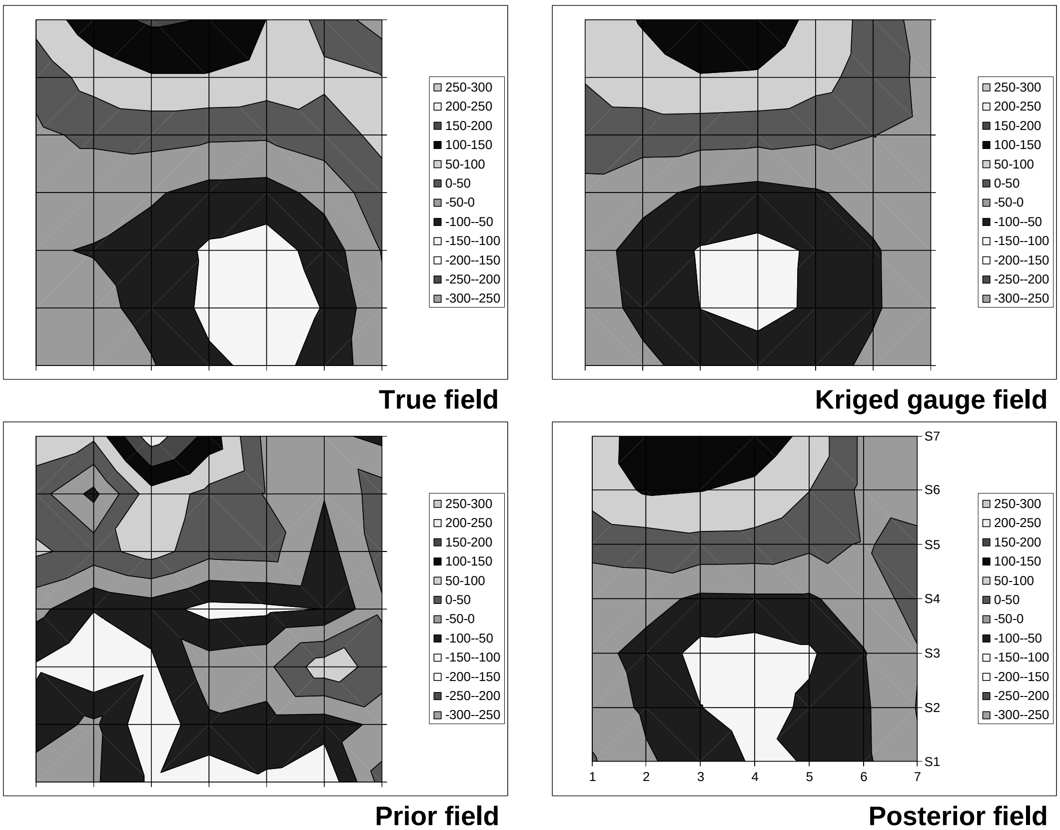


Figure 2.3.12 Plan view of fields representing a single time realization

2.3.3 - A REPEAT OF TODINI'S EXPERIMENT ON A PIXEL BY PIXEL BASIS

The computationally efficient method suggested in Section 2.3.1 was tested on the same set of 2000 random fields to compare the results to those achieved by the full field Bayesian merging technique. The results are presented and compared in terms of bias and variance reduction from the noisy "radar" estimates of the "true" field.

Figures 2.3.13 to 2.3.15 show the equivalent comparisons to Figures 2.3.10 to 2.3.12.

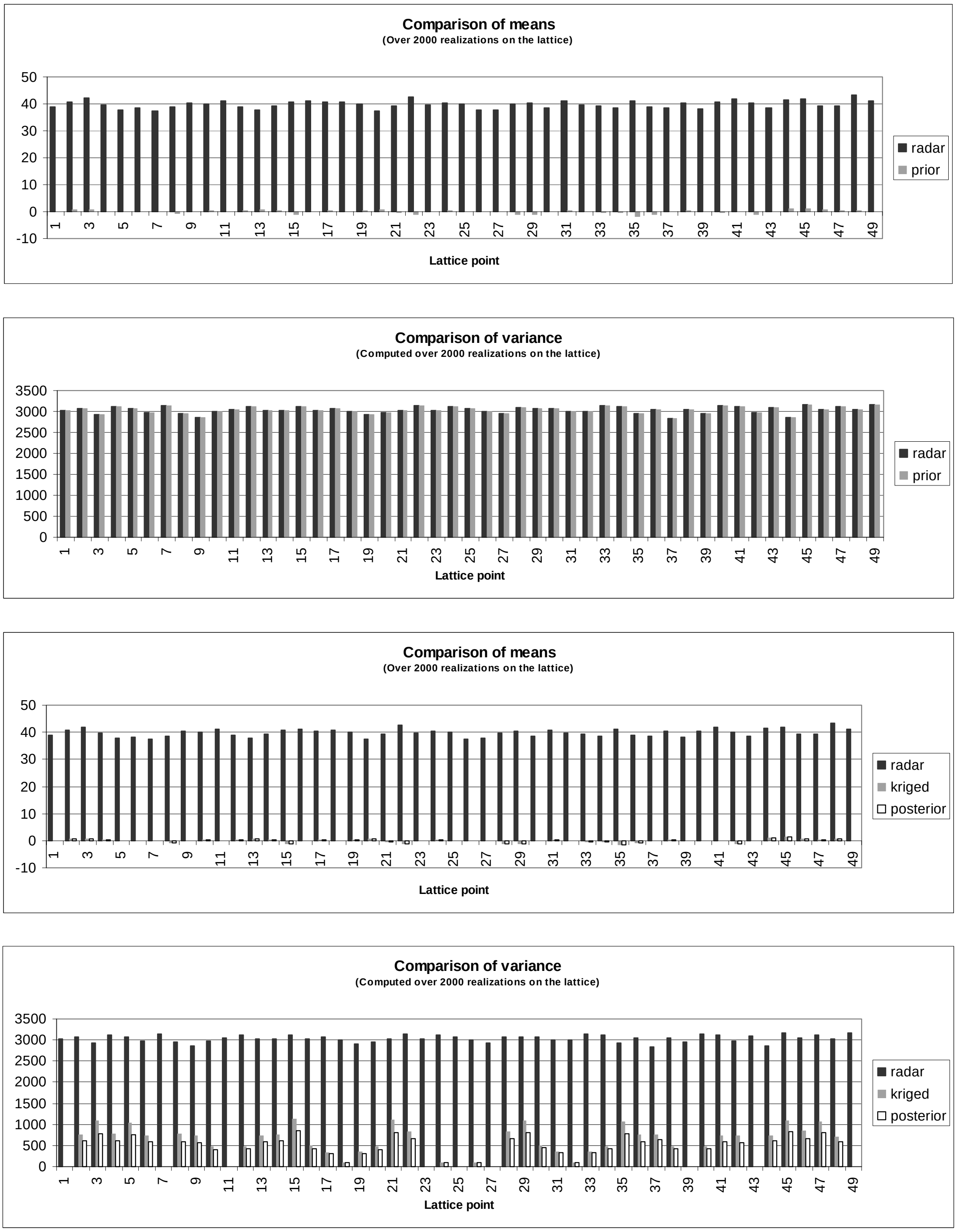


Figure 2.3.13 Comparison of statistics of residuals

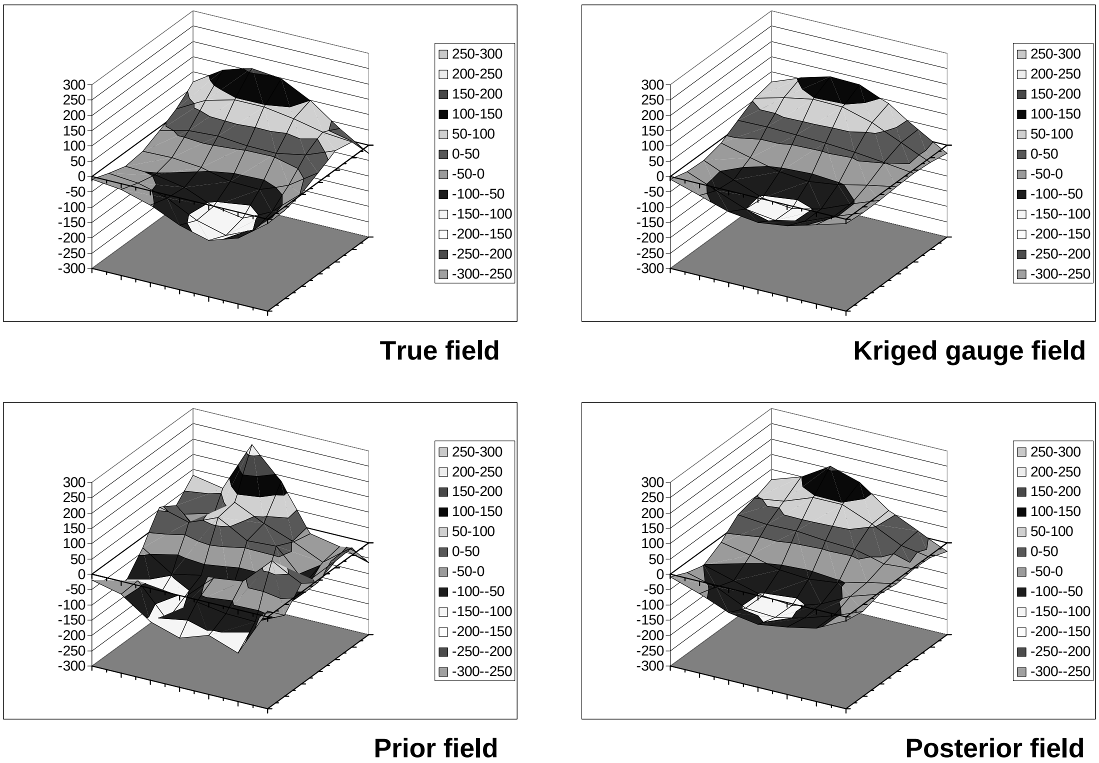


Figure 2.3.14 3D view of fields representing a single time realization

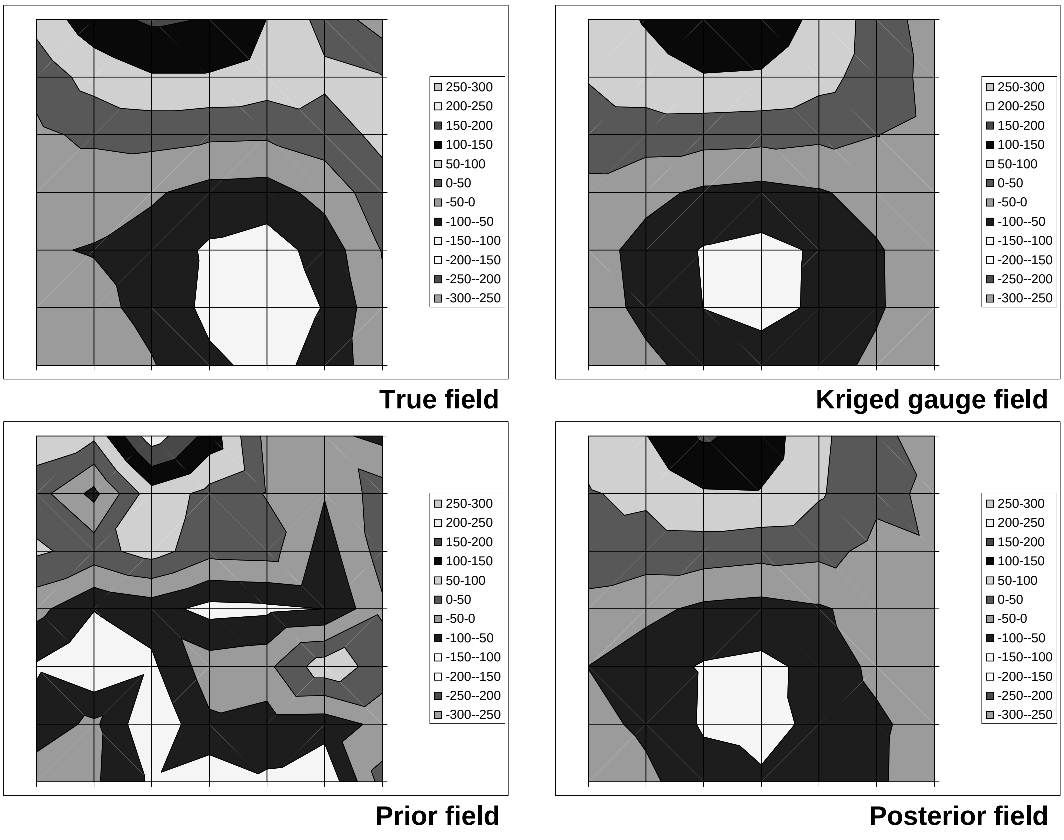
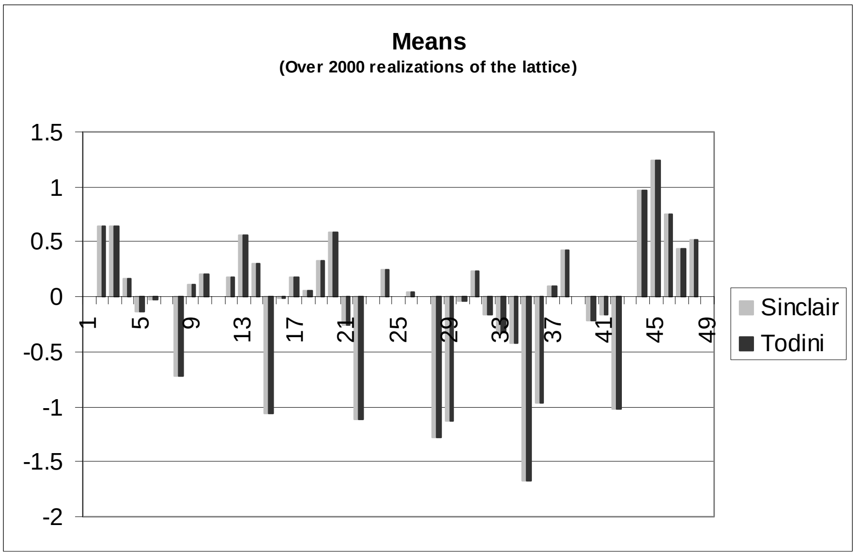


Figure 2.3.15 Plan view of fields representing a single time realization

Figure 2.3.16 shows a comparison of statistics of the posterior residuals for each method. It is clear that the means on each lattice square are identical while the variance on each square is very similar for each method. Todini's full method seems to perform no better as far as the means go on most lattice squares but does not further reduce the variances by a significant amount. The use of a larger block size than the 7x7 adopted in this study would perform better.



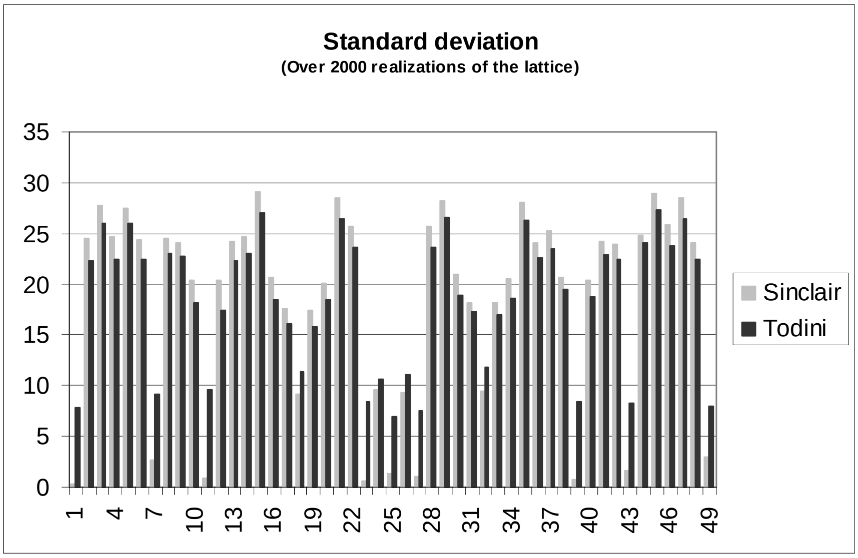


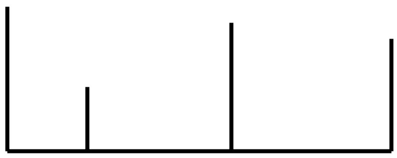
Figure 2.3.16 Comparison of Posterior residuals computed using Todini's (2001) full method and the pixel-by-pixel approximation.

2.3.4 - THE CONDITIONAL MERGING TECHNIQUE

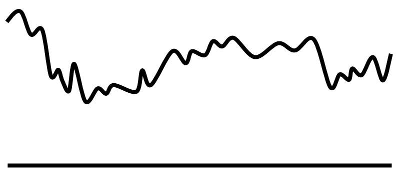
The conditional merging technique of Ehret (2002), introduced by Pegram (2002), also makes use of Kriging to extract the optimal information content from the observed data. The description of the technique (for a one dimensional case) given below is adapted from Ehret's work.

Merging algorithm

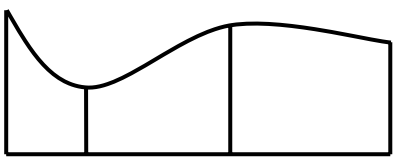
(a) The rainfall field is observed at discrete points from rain gauges.



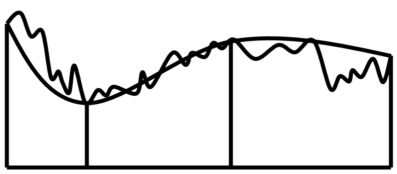
(b) The rainfall field is also observed by radar on a regular, volume-integrated grid.



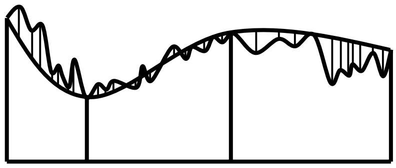
(c) Kriging and the rain gauge observations are used to obtain the best linear unbiased estimate of rainfall on the radar grid.



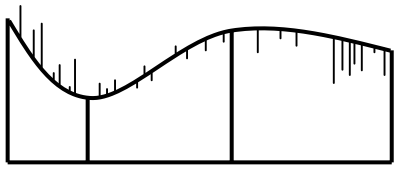
(d) The radar pixel values at the rain gauge positions are Kriged onto the remainder of the radar grid.



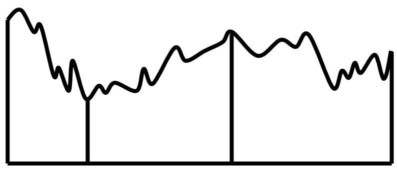
(e) At each grid point, the deviation **C** between the observed and interpolated radar value is computed using a suitable method.



(f) The field of deviations obtained from (e) is applied to the rainfall field obtained from Kriging the rain gauge observations.



(g) A rainfall field that follows the mean field of the rain gauge interpolation, while preserving the mean field deviations and the spatial structure of the radar field is obtained.



The spatial structure of the merged field is therefore obtained from the radar while the rainfall values are "stitched" to the gauge observations of the "true" rainfall field.

2.3.5 - COMPARING THE BAYESIAN AND CONDITIONAL MERGING TECHNIQUES

The experiments reported on in the previous sections were not thought to be representative of true rainfall fields. To test the performance of the merging techniques the following experiment was devised, based largely on the experiment carried out by Todini (2001).

A sequence of 1000 independent 128x128 pixel rainfall fields was produced using the "String of Beads" simulation model (Pegram and Clothier, 2001). These rainfall fields were treated as the "true" rainfall field and "observed" radar estimates respectively of the "true" field, produced by adding bias and noise. The "true" field was sampled at 83 "rain gauge" locations chosen randomly on the pixel grid. This gives an average coverage of one gauge for every 198 km^2, representing an average gauge spacing of 14 km, which is a fairly dense network. The "rain gauge" measurements were assumed to be without error. A single realization of the "true" field as well as the corresponding "observed" radar estimate and the Kriged gauge field are also shown in Figure 2.3.17, while a more detailed description of the experiment follows.

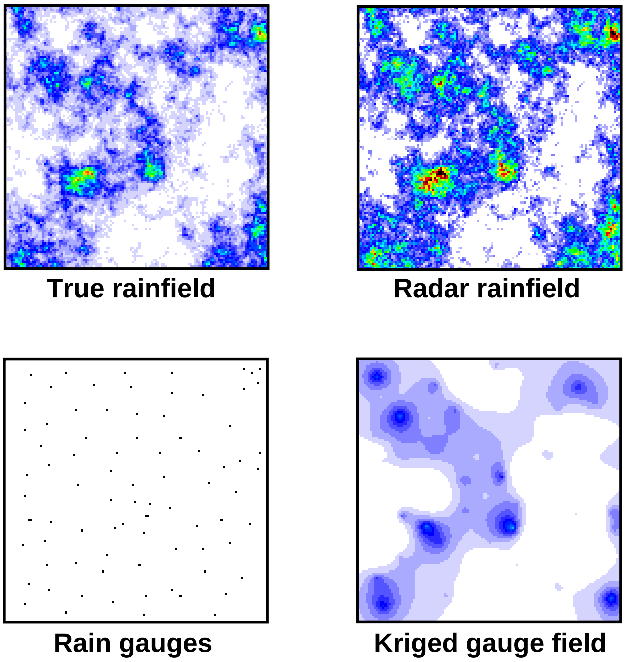
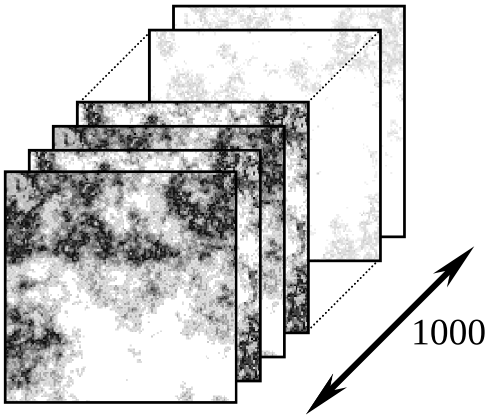
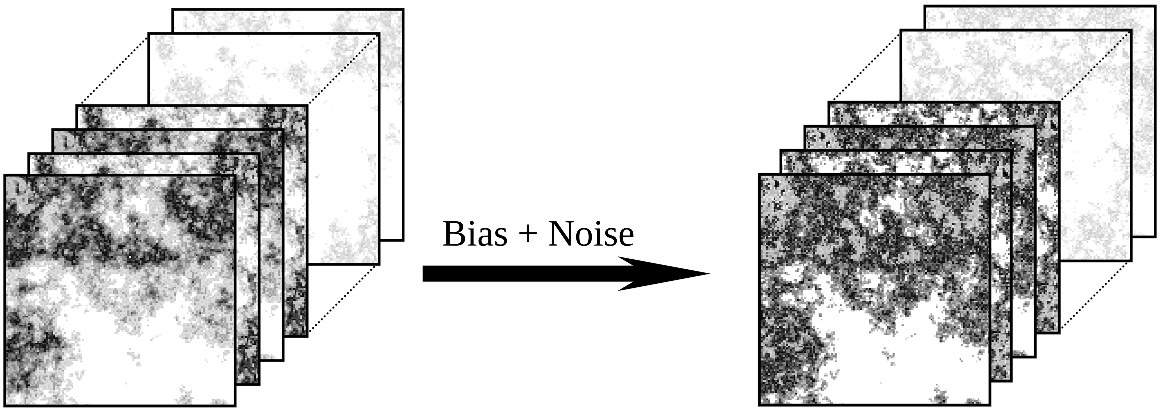


Figure 2.3.17 A single realization of the modelled instantaneous rainfall fields (mm/hr)

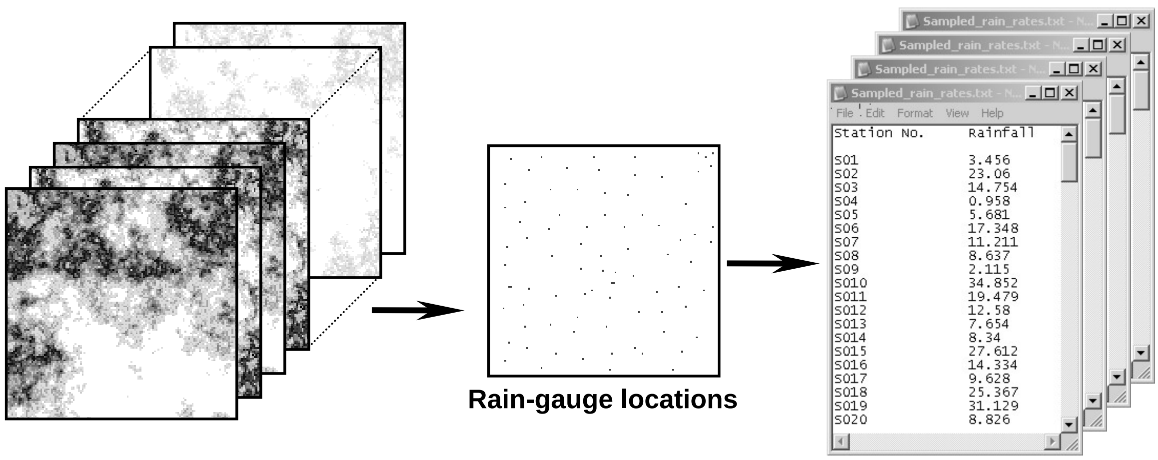
i) Generate 1000 independent rainfall fields using the "String of Beads" model.



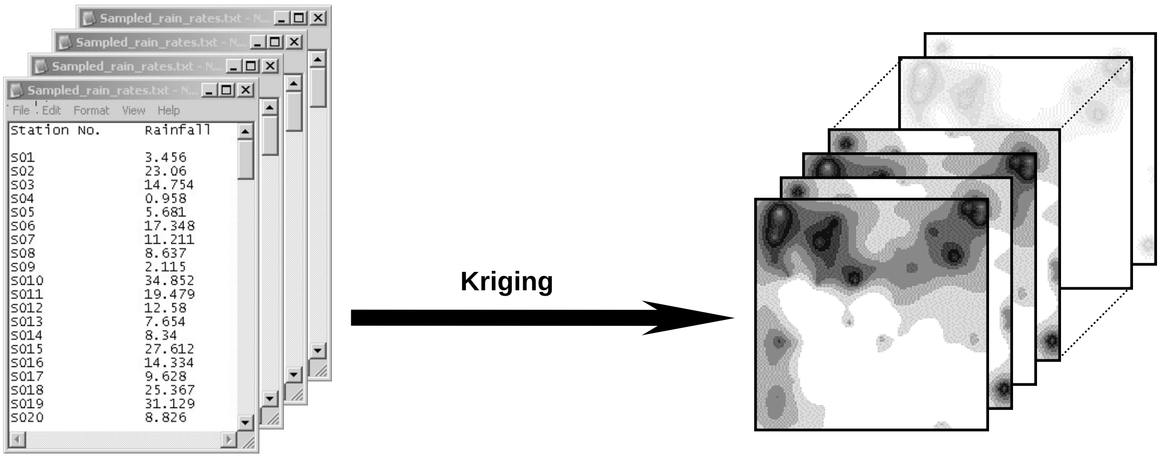
ii) Add bias and noise to simulate radar measurements of the "true" rainfall field.



iii) Sample the "true" rainfall field to get a set of unbiased rain-gauge observations.



iv) Krige the rain gauge observations onto the unobserved radar pixels.



v) Apply either the Bayesian or Conditional merging procedure.

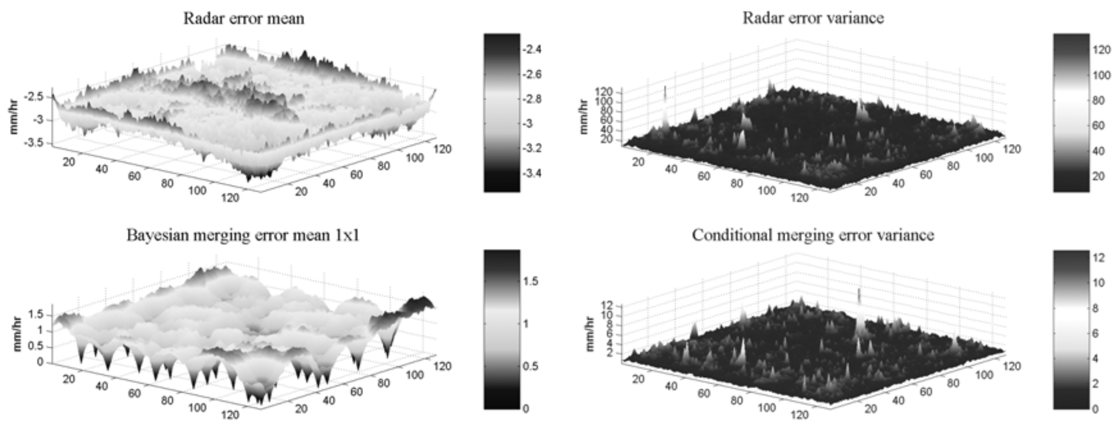
vi) Compute the mean error at each pixel over the 1000 realizations.

$$m.e_{pixel} = \frac{1}{N}\sum_{i=1}^{N}\{true\_value_{pixel}(i) - merged_{pixel}(i)\}$$

vii) Compute the variance of the errors at each pixel over the 1000 realizations.

$$\mathrm{var}_{pixel} = \frac{1}{N}\sum_{i=1}^{N}\{true\_value_{pixel}(i) - merged_{pixel}(i) - m.e_{pixel}(i)\}^2$$

viii) Compare the mean and variance computed from steps vi) and vii) to those computed from the radar errors in order to quantify the "improvement" gained from each merging technique.



A single realization of the true field, radar observation and merged fields is shown in Figure 2.3.18.

Figure 2.3.19 shows histograms of the mean errors on each field. The radar error mean was computed by taking the mean value (over 1000 realizations) of the residuals between the true field and the radar observation at each pixel. This results in the 16384 values plotted as a histogram. The mean errors for each merging technique were computed in the same way. It is clear from the figure that each technique is an improvement over the radar observation and that the Conditional merging technique performs somewhat better. The spike of values at zero in each case is due to the 83 rain gauge observations that are of course, perfect and without any bias.

The mean variance of the errors is reported in a similar manner in Figure 2.3.20. Once again both merging techniques represent an improvement with the Conditional merging technique performing better in terms of variance reduction. The choice of merging technique for this study was therefore the Conditional merging method based on the results presented above.

The results of this comparison between the Bayesian method and the Conditional merging method were reported at the EGS/AGU/EUG Assembly by Sinclair et al. (2003).

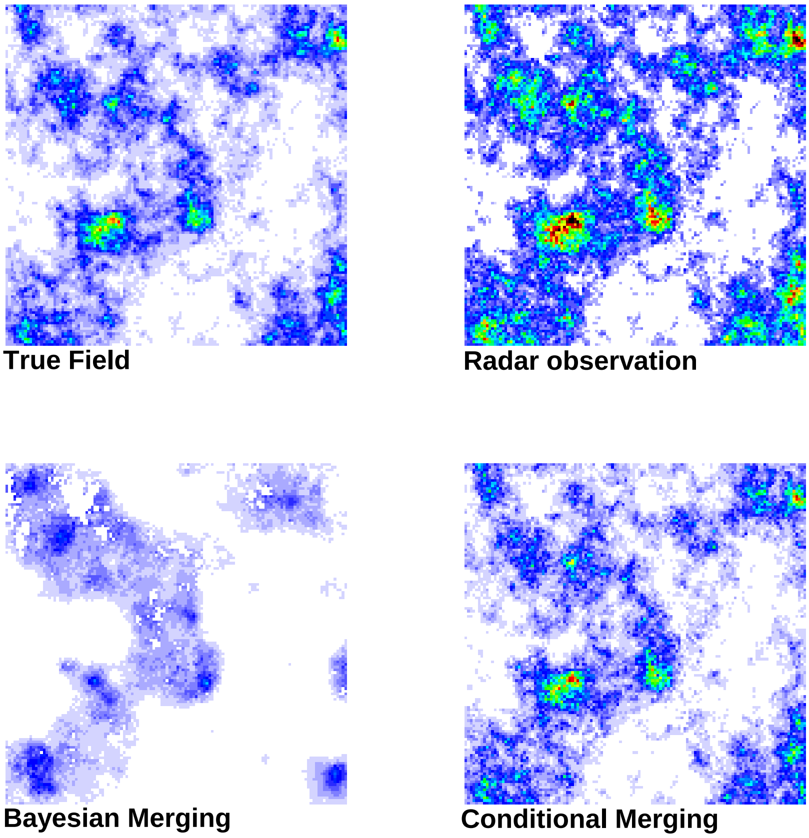


Figure 2.3.18 Comparison of merged fields for a single realization

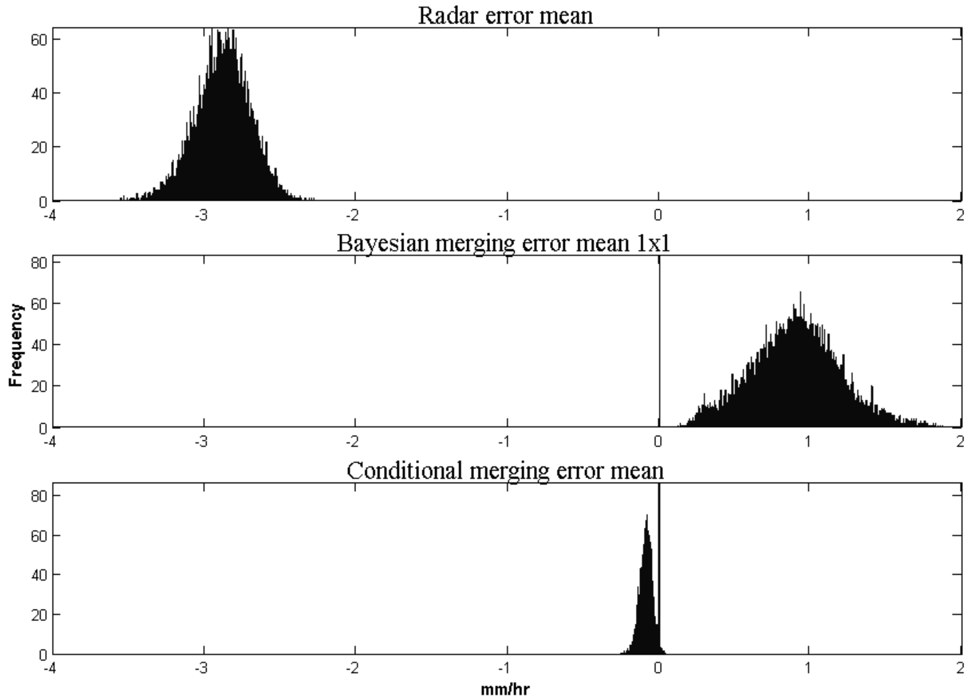


Figure 2.3.19 Reductions in mean errors from each merging technique

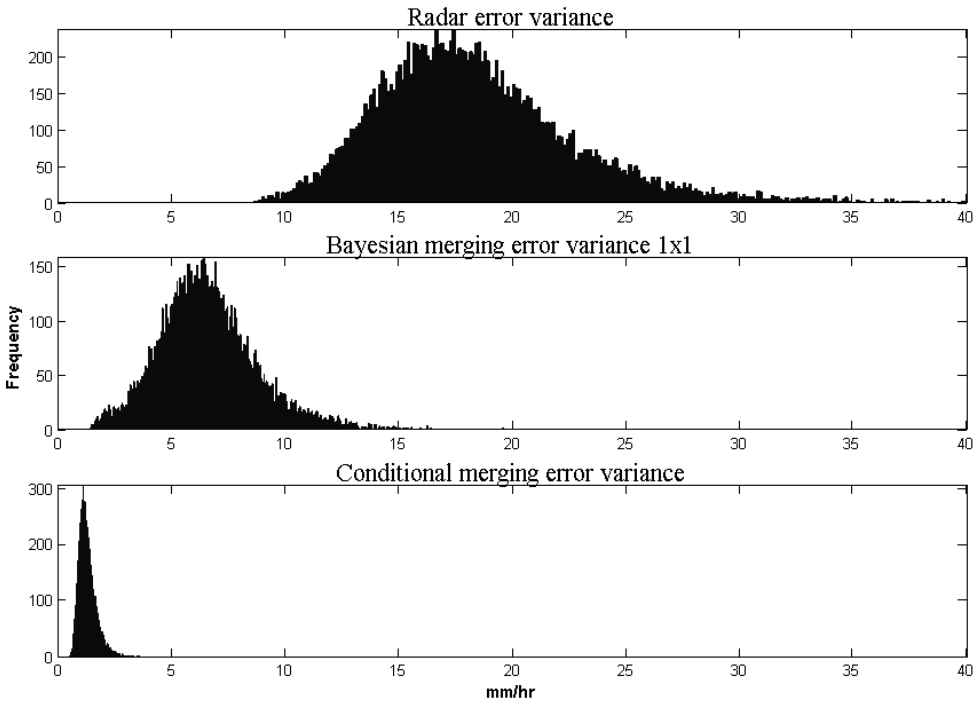


Figure 2.3.20 Variance reductions achieved by the merging techniques

2.4 - SHORT TERM RAINFALL FORECASTING

Forecasts of rainfall can be achieved by applying one of two broad categories of model; these are physically based models and statistically based models. The class of physically based models attempts to model the complex physics of the atmosphere, which would be an ideal solution if computing resources allowed the models to make computations at the spatial and temporal resolution required to properly characterise the variable nature of rainfall. Global circulation models (GCM's) are only able to perform operational simulations at synoptic scale spatial resolutions with a time step of about a day. Nesting limited area models (LAM) in a GCM can provide greater spatial detail up to resolutions of a few (60x60 km) kilometres (Engelbrecht *et al*, 2002), it is hoped that advances in computing power will allow such models to provide simulations and forecasts at higher resolutions.

Statistical models are models which try to match certain key statistics of the rainfall as measured by a particular instrument; for example an application of the Neyman-Scott shot noise model presented by Cowpertwait (1991) tries to model the arrival times and duration of rainfall events as measured by rain gauges, while the "String of Beads" model (Pegram and Clothier, 2001) models the spatial and temporal characteristics of rainfall fields measured by radar. For the reasons described in Section 2.3 (advocating spatial rainfall fields as input to the catchment model) a decision was taken to limit this study to models capable of modelling spatial rainfall fields rather than considering point process models.

Rainfall forecasts applicable to flood forecasting (ideally) span a whole range of timescales from seasonal outlooks through several days ahead, right down to forecasts for the next few minutes or hours. In order that the longer range forecasts are significant, the uncertainties associated with them need to be reduced by decreasing the spatial resolution (increasing the basic area from pixel-scale) at which the forecasts can be made. This unfortunately has the effect of reducing their usefulness for flood forecasting, even for relatively large catchments. For this reason two statistically based models were investigated as possible candidates to provide short term forecasts of rainfall fields in this study. These were the "String of Beads" Model (SBM) (Pegram and Clothier, 2001 and Clothier and Pegram, 2002) and the "Spectral prognosis" (S_PROG) model (Seed, 2001). Numerical weather prediction is not able to provide sufficient quantitative detail on the appropriate spatial and temporal scales for the purposes of this project (Barron *et al*, 1999).

2.4.1 - FORECASTING WITH THE "STRING OF BEADS" MODEL

Developing a forecasting implementation of the "String of Beads" rainfall field simulation model proved to be more difficult than had originally been anticipated. The process resulted in some improvements to the model, particularly to the advection routines. A complete description of the SBM in simulation mode is given by Clothier and Pegram (2002), with a basic outline given here to illustrate the differences between SBM in forecast mode and in simulation mode.

A typical simulation begins with the generation of a stack of random noise fields. The values of the noise at each pixel in the field are independent and distributed according to the standard normal distribution with a mean of zero and unit variance. A new field is constructed (pixel by pixel) using an Autoregressive lag 5, AR(5), model and placed at the first position on the stack, with the existing fields moving backwards one position and the final field falling away (Figure 2.4.1).

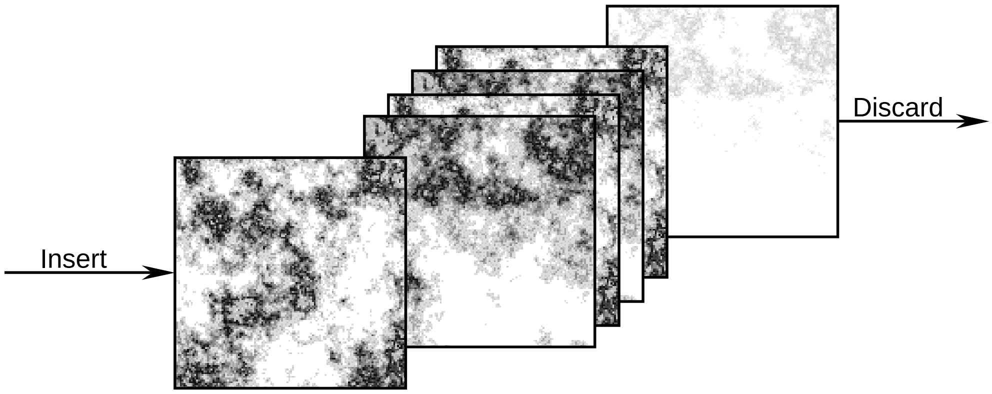


Figure 2.4.1 The SBM noise stack

A mean field advection vector is used to establish the correct temporal alignment of the pixel values in the field. A warm up period is required to ensure that the sequence of fields used in the generation is properly conditioned. Once the stack has been given a sufficient number of recursions to be correctly conditioned, the newly generated field is power law filtered to increase the spatial correlation to the desired level, then the correct image scale statistics “Wet Area Ratio” (WAR) and “Image Mean Flux” (IMF) are imposed on the field by a thresholding and scaling process. The resulting field is then exponentiated to produce a field of simulated rainfall rates.

In forecast mode the process is more streamlined. The spatial correlation structure from the observed fields is retained and the pixel scale development, now using an AR(2) model, computed directly from the observed fields. The image scale statistics are forecast using the same bivariate AR(5) process of the simulation mode but this time conditioned on the values from the previous five observations. Additionally a sophisticated motion-tracking algorithm (Bab-Hadiashar et al., 1996; Seed, 2001) is used to estimate the field advection. A dense grid of advection vectors is computed at each time step and used to advect the forecasts and maintain the appropriate temporal alignment between forecasts. The smoothed advection grid is updated, at each time step, as new information becomes available. The computation of the advection vectors is efficient enough to allow it to be used for real-time applications. Optical flow algorithms that provide comparable (or better) accuracy are generally prohibitively inefficient to compute without specialized hardware components (Camus and Bulthoff, 1995).

2.4.2 - THE S_PROG MODEL

The S_PROG model exploits the idea that rainfall fields exist as structures encompassing a range of spatial scales with the persistence of structures being proportional to their spatial scale i.e. larger scale structures have a longer persistence time than smaller scale ones (Seed *et al*, 1999). Generation of new forecast rainfall fields is achieved by first disaggregating the observed field into a multi-layered hierarchy of fields each of which represents features at different scales. This disaggregation is achieved by applying a notch filter in the Fourier domain to separate the field into a number of spectral components. Each of the component fields in the hierarchy is modelled by a dynamically fitted AR(2) process evolving at the relevant space scale, and combined to produce the forecasts. The forecast field is conditioned to have the same mean rainfall rate and wetted area as the latest observed image, this is in contrast to the SBM where the evolution of these quantities is predicted using a fixed bivariate AR(5) model.

2.4.3 – FORECAST MODEL COMPARISONS

Both qualitative and quantitative comparisons were made between the forecasts of SBM and S_PROG. Both models were configured to produce forecasts conditioned on an observed initialisation period of a rainfall event measured by the Durban weather radar and the resulting forecasts compared to the observations.

Figures 2.4.2, 2.4.4, 2.4.6 and 2.4.8 show examples of forecasts made using the SBM and S_PROG models as well as the observed sequences from the equivalent times. The forecasts have been made at time steps of approximately 5 minutes, thus the forecasts in Figure 2.4.2 are up to half an hour ahead. The images shown in the figure represent observed and forecast reflectivity fields measured in dBZ. All fields have been thresholded below a value of 10 dBZ because the rainfall rates below this level were considered to be negligible (approximately 0.15 mm/hr). The forecast fields were compared to the observed fields in terms of two measures. The first was the square root of the mean of the sum of squared errors over all pixels in the field (RMSSE) while the second was a comparison of the mean absolute error, this is closely related to the IMF in the rainfall rate domain. Figures 2.4.3, 2.4.5, 2.4.7 and 2.4.9 compare the magnitude of these two measures for a set of four different forecast sequences produced by the models.

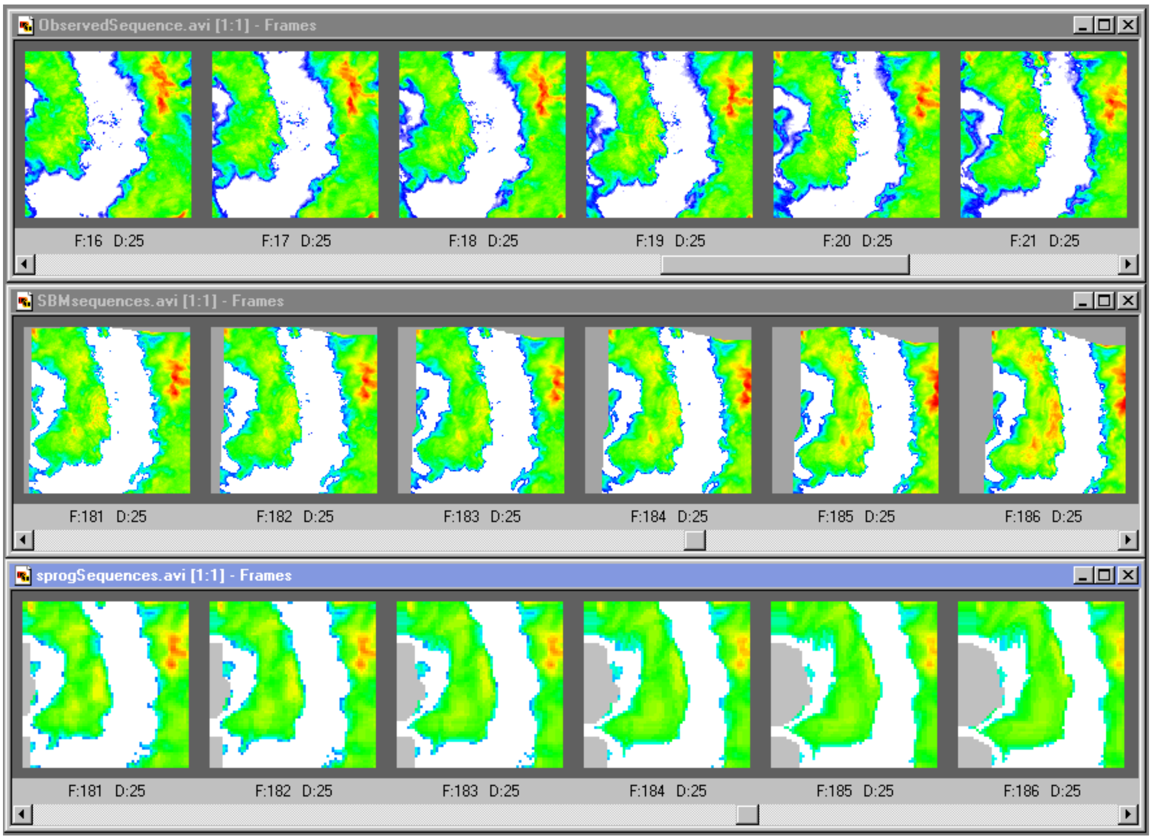


Figure 2.4.2: Comparative forecasts and observed reflectivity fields (Sequence 1) – top row Observed, second row SBM forecasts, third row S_PROG forecasts - 5 minute intervals starting simultaneously.

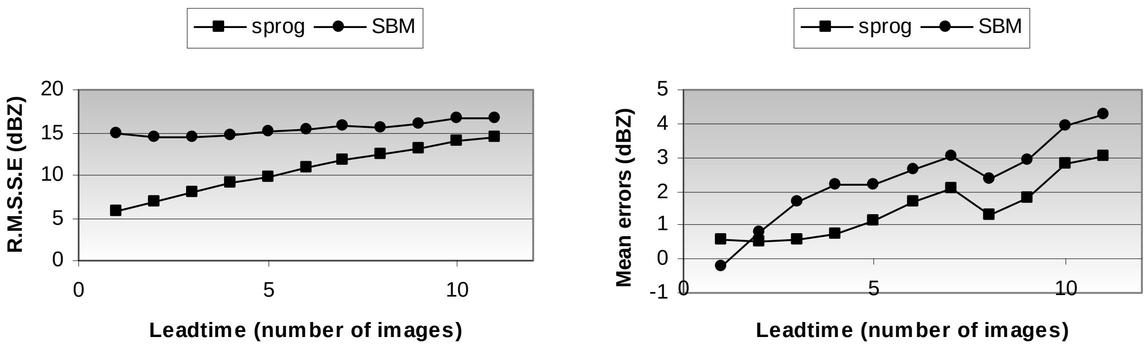


Figure 2.4.3 Root mean sum of squared forecast errors and mean errors (Sequence 1)

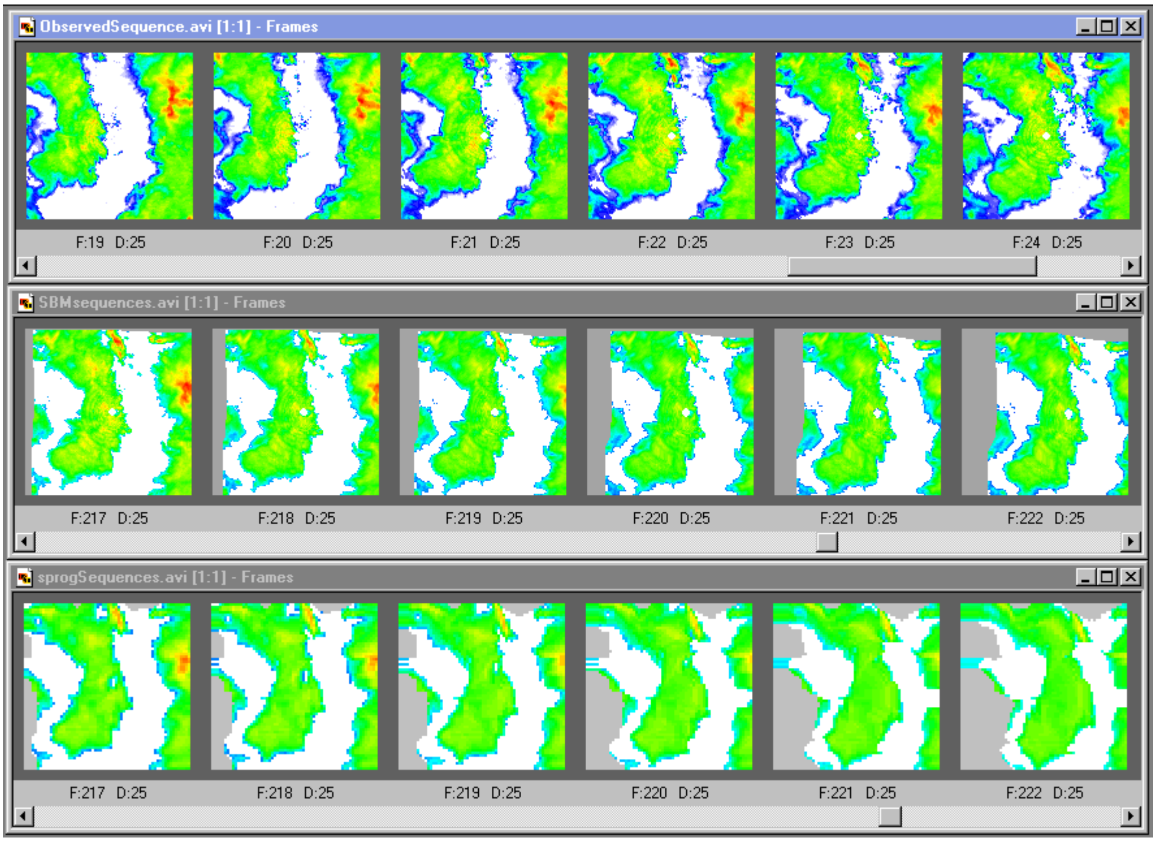


Figure 2.4.4: Comparative forecasts and observed reflectivity fields (Sequence 2) – top row Observed, second row SBM forecasts, third row S_PROG forecasts - 5 minute intervals starting simultaneously.

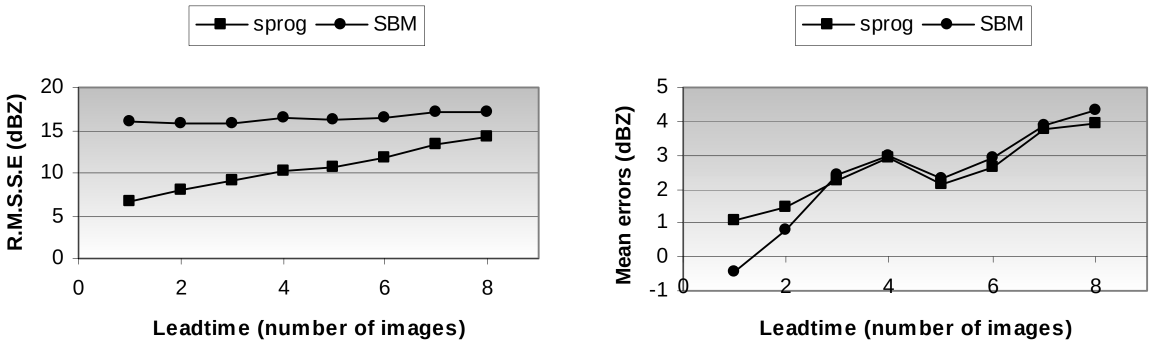


Figure 2.4.5 Root mean sum of squared forecast errors and mean errors (Sequence 2)

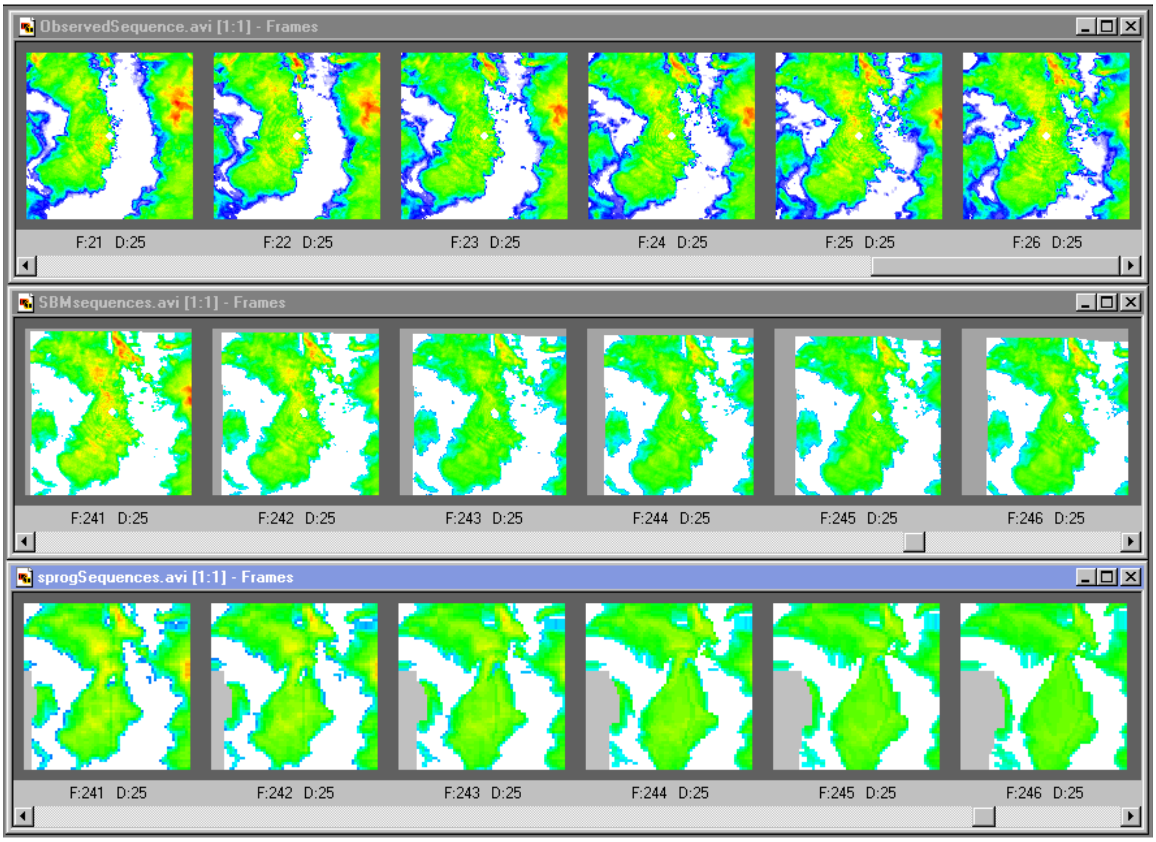


Figure 2.4.6: Comparative forecasts and observed reflectivity fields (Sequence 3) – top row Observed, second row SBM forecasts, third row S_PROG forecasts - 5 minute intervals starting simultaneously.

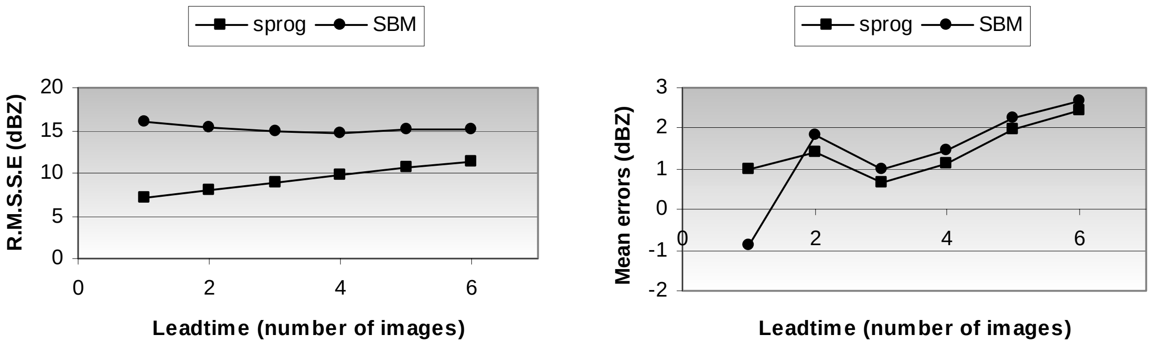


Figure 2.4.7 Root mean sum of squared forecast errors and mean errors (Sequence 3)

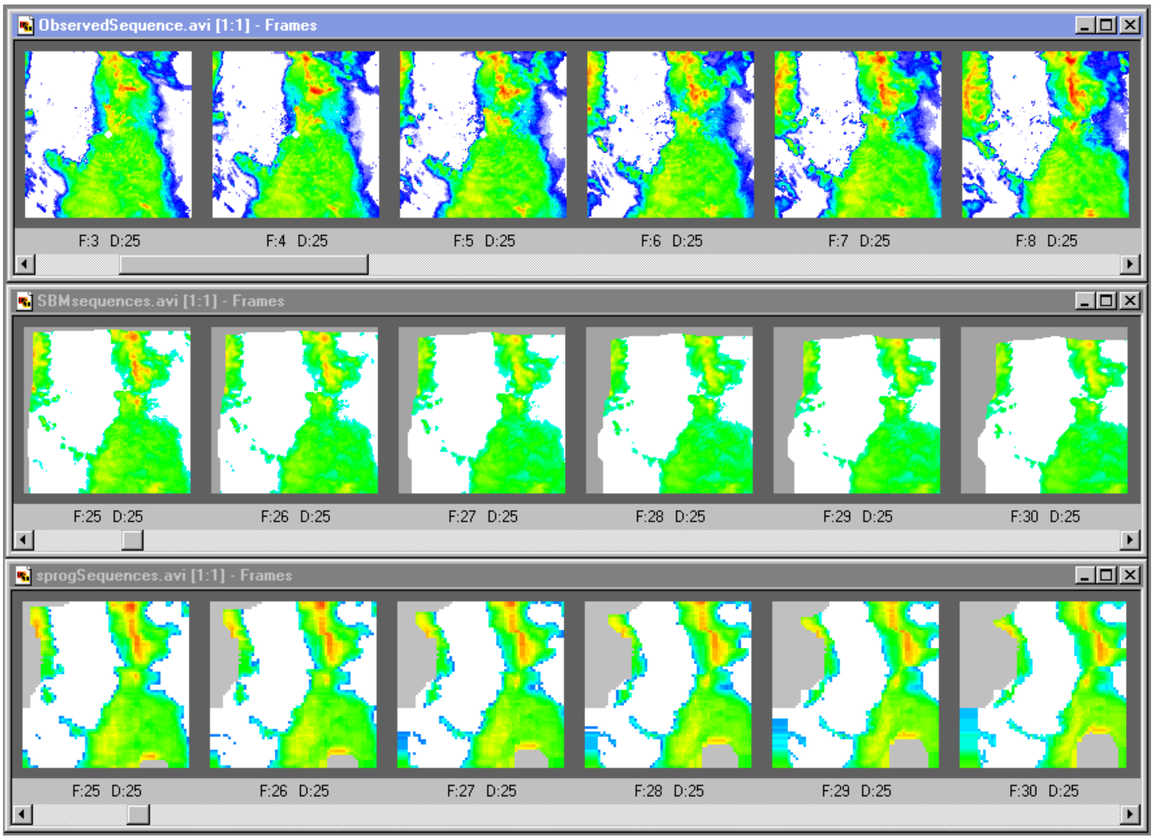


Figure 2.4.8: Comparative forecasts and observed reflectivity fields (Sequence 4) – top row Observed, second row SBM forecasts, third row S_PROG forecasts - 5 minute intervals starting simultaneously.

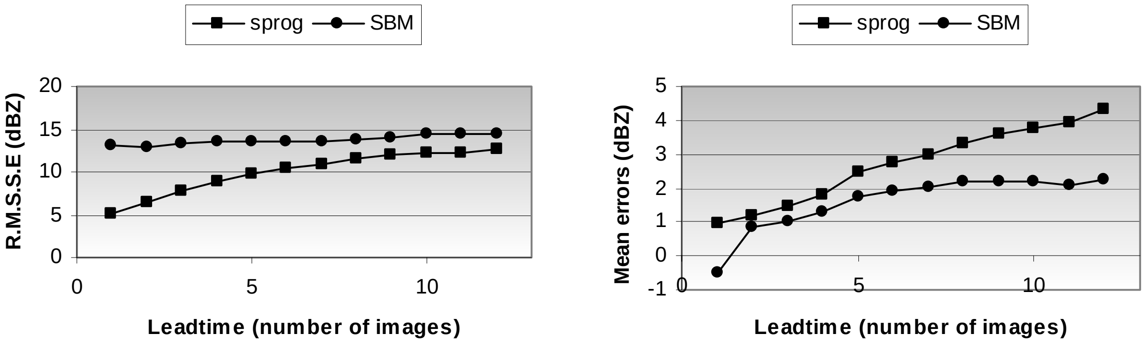


Figure 2.4.9 Root mean sum of squared forecast errors and mean errors (Sequence 4)

S_PROG performs better than SBM in all cases if RMSSE at the primary pixel scale is the only criterion used for the comparison. The RMSSE for S_PROG increases rapidly with forecast time while the RMSSE for SBM remains fairly constant with time, despite starting at a higher initial value. It makes sense that S_PROG performs better (in this case) since the model is formulated to give an optimum forecast in the least squares sense, with the forecasts being degraded to a mean field as the forecast time increases resulting in errors with a smaller magnitude. The SBM does not behave in this way since the image scale parameters (WAR and IMF) evolve according to the bivariate AR(5) model meaning that changes in intensity and wetted area are evident in the forecasts. The RMSSE is very sensitive to the errors due to the relative positioning of peaks in the field. Squaring the error values removes the effects of sign from the comparison. Figure 2.4.10 shows this effect graphically for a single dimension. 2.4.10a shows the arrival times of three observed rainfall amounts while 2.4.10b shows the forecast arrival times, the magnitude of the rainfall is exactly correct but the arrival times of the pulses are not correctly forecast. Figure 2.4.10c shows the result of taking the difference between 2.4.10a and 2.4.10b. It's obvious from this that the mean error will be zero since the rainfall volume is the same in both cases. However, 2.4.10d shows the squared errors and, again, it's clear that the sum of squared errors will be non-zero. In this case the RMSSE will be larger (a poorer "fit" to the observations) than the RMSSE that would result from making a forecast that expects no rainfall at all. It is for this reason that the RMSSE can be considered misleading and the mean absolute errors were also considered in the model comparison.

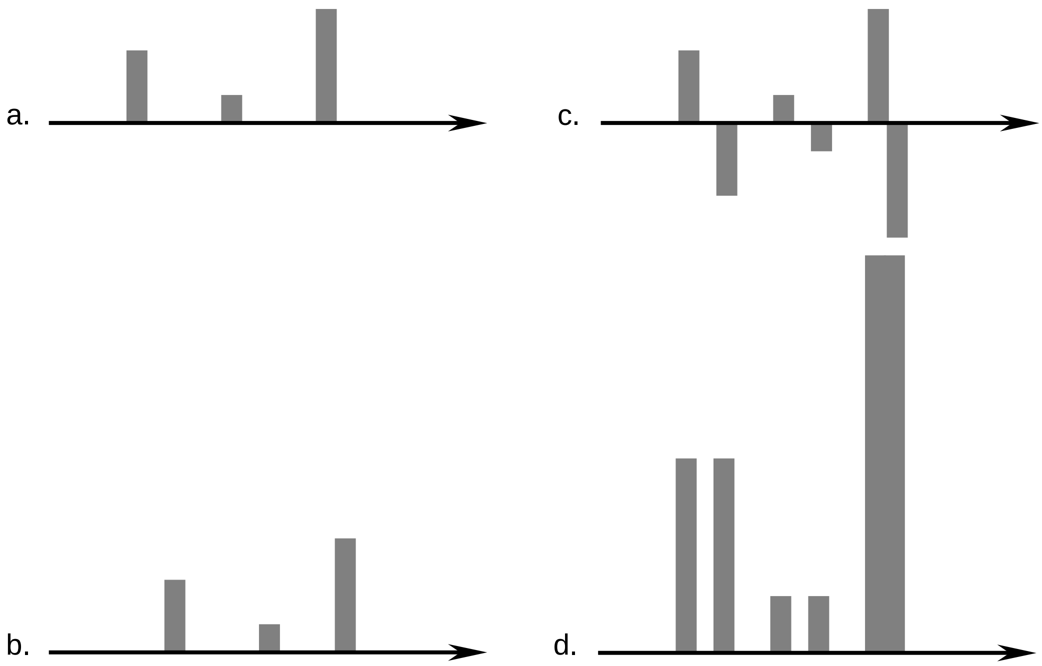


Figure 2.3.10 Sensitivity of RMSSE to the position of peaks

The eventual decision was to use S_PROG as the forecasting model since SBM is better suited to event scale simulation at this stage. Further development of the bivariate AR(5) process is expected to make ensemble forecasts possible with SBM, however this work was considered to be beyond the scope of this project.

2.5 - AREAL RAINFALL ESTIMATES

The observed and forecast spatial rainfall estimates need to be converted into a suitable form for input to a rainfall-runoff model of the catchment being considered. For this purpose the hourly rainfall accumulations are averaged over each of the WR90 quaternary catchments comprising the Mgeni and Mlazi catchments. Spatial averaging is achieved by masking the appropriate parts of the accumulated rainfall fields as described in Pegram and Sinclair (2002).

The first step is to **accumulate the spatial rainfall fields** over the required time period (usually an hour). This can be done in a number of ways with the choice of method having a significant effect on the resulting accumulated fields. For the purposes of this project a linear superposition accumulation scheme was adopted. However, this is not necessarily the most appropriate option for accumulating rainfall caused by small scale and fast moving rain systems. The rainfall fields are sampled, instantaneously, at discrete time-steps, while the true rainfall field is evolving continuously with time. This evolution encompasses the growth and decay of the field's structures as well as complex field advection processes. If the sampling interval is not short enough to capture the dynamic changes in the field then simple linear accumulation techniques are inadequate and more sophisticated accumulations schemes are required.

Figure 2.5.1 shows a 5 hour 50 minute accumulation done using a linear superposition technique (where the time between radar scans is 15 minutes) and Figure 2.5.2 the same accumulation period using a more sophisticated approach described by Hannesen (2002b). It is obvious that the second approach produces an accumulated field that is more consistent with the way we would expect a rainstorm to move across the radars field of view. Future developments in this area will improve the spatial description of rainfall accumulations.

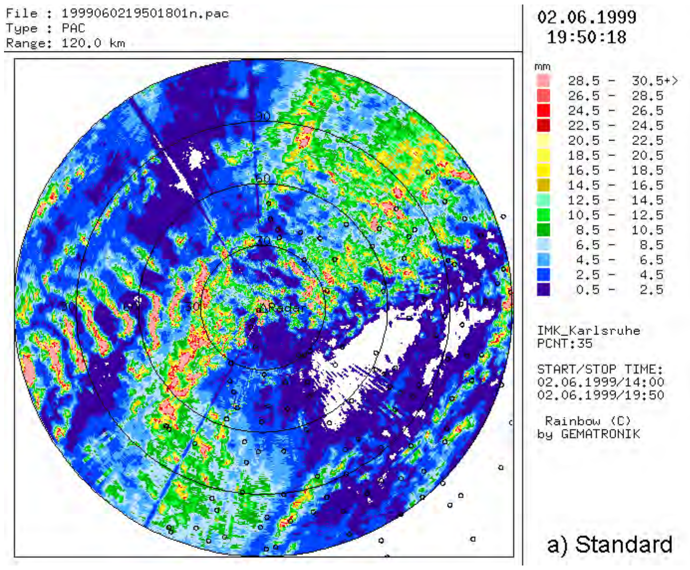


Figure 2.5.1 A superimpositional accumulation technique (after Hannesen 2002b)

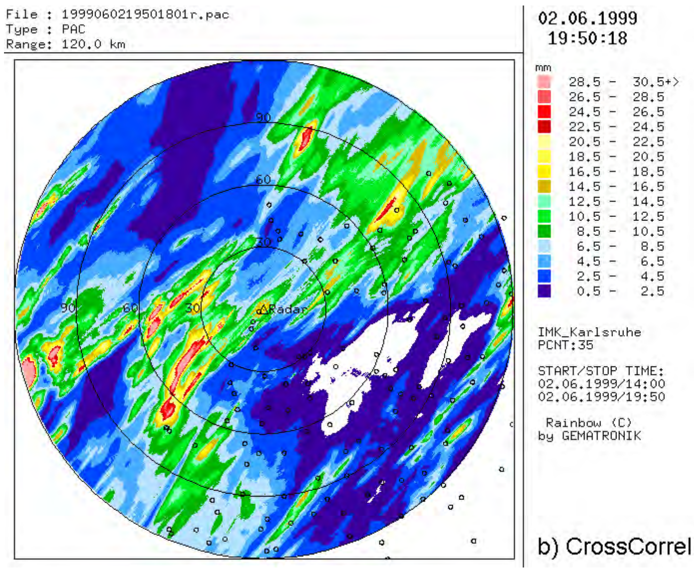


Figure 2.5.2 Cross-correlation tracking and morphing accumulation method (after Hannesen 2002b)

2.6 - DISSEMINATION OF OUTPUTS

The dissemination of outputs refers to the kinds of information presented to the end user of the flood forecasting system, in this case the disaster managers (DMs). The products were agreed upon after consultation with the members of the disaster management team at eThekwini Metro. The role of the flood forecasting system in the process of disaster management is to provide a warning of the possibility and seriousness of an impending flood event, the actions and decisions taken on the basis of that information are then left up to the relevant authorities. The consultation process yielded two major requirements from the DMs point of view; some warning should be provided ahead of a likely flood and the affected areas should be indicated (preferably in a graphical format). To meet these requirements, the Arcview GIS was chosen as a display tool since the DMs already make use of this system. Images of current and historical rainfall are available for display in the disaster management control room and dynamically selected flood lines may be called up in response to current observed and forecast streamflows.

2.6.1 – INTEGRATION OF THE RADAR RAINFALL IMAGES INTO A GIS

When radar rainfall images are presented in a spatial context they can provide an advance warning of heavy rainfall approaching sensitive (flood-prone) catchments. Figure 2.6.1 shows an example of an instantaneous radar image of a major storm event that moved over the Mlazi catchment and the lower reaches of the Mgeni catchment.

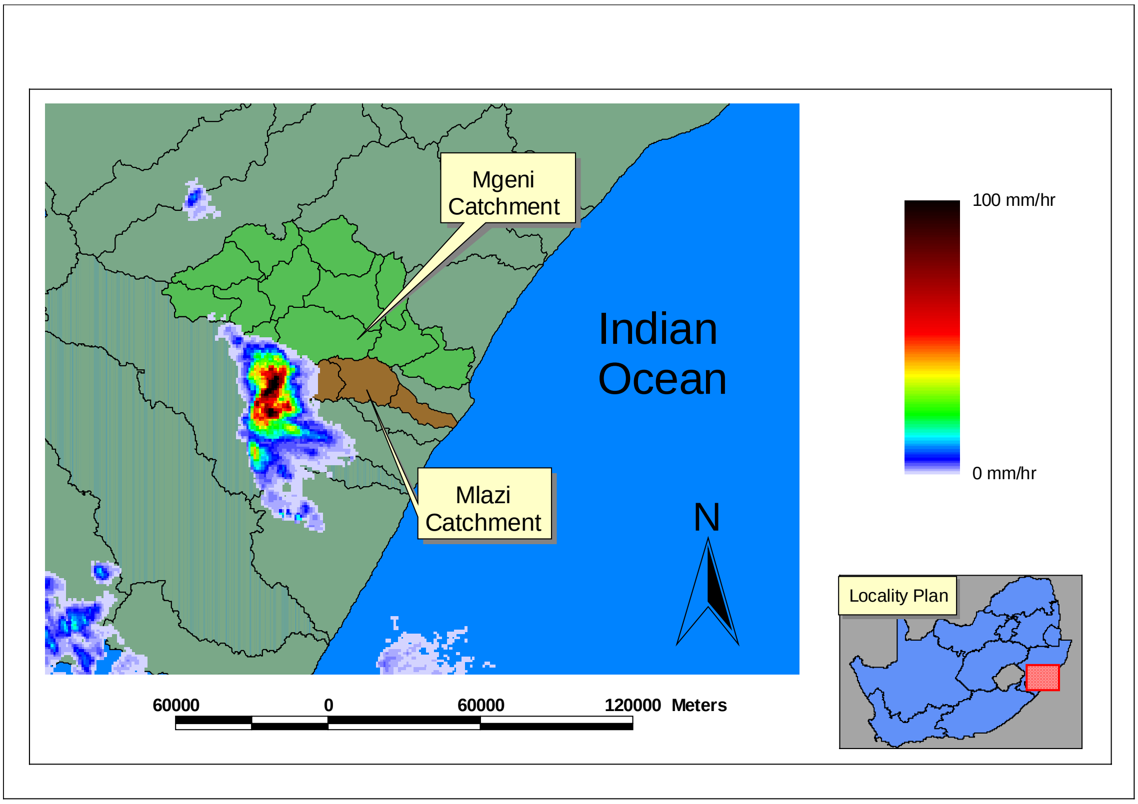


Figure 2.6.1 A major storm over the upper Mlazi catchment (17 December 2002 at 14:58)

The images are available in an Arcview compatible format at the eThekwini municipality's disaster management centre in near real-time. The data transfer process and relevant software was developed in line with the project objectives specified in section 1.3 and the transfer methodology is briefly described in the following paragraphs.

The SAWS make use of the "Meteorological Data Volume" (MDV) file format to transfer and archive spatial radar data. In order to make use of this data-set it is necessary to have a means of extracting the data from MDV files.

The C language source code for MDV file handling was obtained from METSYS and compiled under Microsoft Windows as a set of statically linked libraries. Linking to these libraries allows the user to make use of the MDV file handling routines in custom software used to extract and manipulate the MDV data. A full understanding of the MDV routines has been achieved and they are now an integral part of many of the software products developed during the course of this study. The version of the libraries being used is different from (older than) the latest version currently in use by METSYS. An upgrade would be rather time consuming and the benefits of doing so are not sufficient to warrant spending this time. The newer libraries, under Linux, are linked when the MDV processing software is compiled at METSYS. The older versions are used mainly for initial testing.

The raw radar data-stream is processed into MDV format at the radar site and transferred to the METSYS server in Bethlehem where SAWS publishes data products on their web page and archives the data. The products currently available on the web page do not provide the powerful data analysis capabilities of a GIS. Software developed during this project is run routinely on the METSYS server. The software accepts the MDV format data as input and produces radar rainfall images in a spatially referenced format which is compatible with the Arcview GIS. The data is compressed to reduce its size and stored on the METSYS server in a buffer which only stores data for 2 days before deleting it. There is no need to preserve this data as the products can be extracted from the archived MDV data if required.

Software developed in this project using the Python language runs at the Durban Metro DM offices to retrieve the latest images from METSYS. The software makes use of the File Transfer Protocol (FTP) to access the METSYS server remotely via the Internet at five to ten minute intervals. The METSYS FTP server is queried for new data at frequent intervals and when new data is found it is downloaded to the client machine; data that resides on the client but is no longer on the server is automatically deleted (but could be stored if required).

2.6.2 – INDICATING THE FLOOD AFFECTED AREAS IN A GIS

There was a need expressed by the DMs to determine which areas they can expect to be affected by floodwaters. It's all very well having a sequence of observed streamflows and possible forecasts of the future flows but without these numbers being interpreted in terms of flood levels, and depths of inundation, they are of no real significance to the DMs. It is the process of translating flows into inundation that is dealt with in great depth in the second volume of this report "Mlazi nowcasting to include depths of inundation". The details of the methods used for the determination of flood lines and inundation depths will not be repeated here. Instead a brief description of the process and the application of these products in the flood forecasting system are presented.

It will be obvious from volume two of this report that the computation of floodlines is a fairly laborious process which, despite the proliferation of software packages, requires a large degree of human interaction. This does not encourage online computation of current and forecast inundation depths as a first choice of *modus operandi*. While this option was investigated and found to be a real possibility it turned out that time constraints made it an infeasible solution to the problem at hand.

A possible process for producing floodlines is as follows. A Hydraulic model of the river channel and adjacent flood plains must be produced, using a Digital Elevation Model (DEM) in combination with field surveys. This is a once-off procedure since the same model may be used to model the effects of many different flow rates. The hydraulic model used in this study was the well-known HEC-RAS model (Brunner, 2001). HEC-RAS routes the flood-wave along the modelled channel and computes flood levels at each model cross-section. A typical output from HEC-RAS (steady state) is shown in Figure 2.6.2. The model can also output the flood levels as a set of points in a three-dimensional co-ordinate system as a text file. In order to interpret these points as flood lines and inundation levels an interpolation between channel cross-sections is required preferably in conjunction with information from a DEM. A polygon can then be produced in a suitable GIS format for viewing in the disaster management centre. In this case an Arcview shapefile like that show in Figure 2.6.3 needs to be produced. A separate floodline must be produced for each flow rate considered.

For the purposes of this pilot study a simple alternative solution was proposed. Floodlines at various recurrence intervals are available in the Arcview shapefile format for the Mgeni and Mlazi rivers through studies commissioned by the eThekwini municipality. These served as the starting point and an Arcview script was written which selects the appropriate floodlines to display based on the current observed streamflow and the forecast flow. Although the flood-wave is dynamic, it is assumed (for steep channels with negligible off-channel storage) that the inundation levels traced by the flood peak will be closely approximated by the corresponding steady-state peak, whose inundation level is the dynamic wave's upper bound. At the time of writing this system is not operational due to technical delays in obtaining real-time flow information. However, it is due to be implemented at the eThekwini municipality in the very near future as a deliverable of the follow-on project to this pilot study.

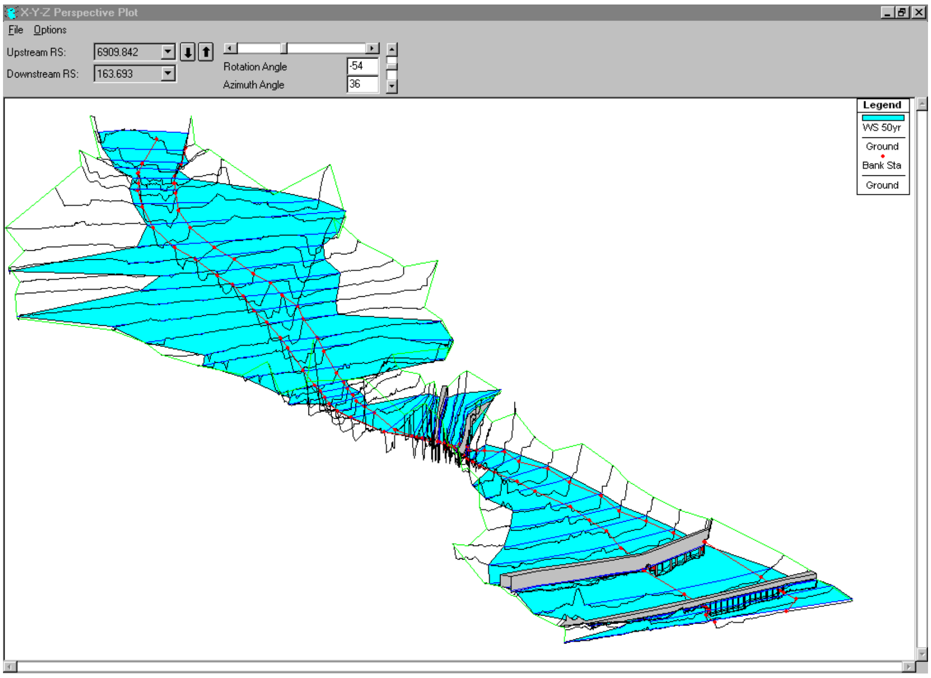


Figure 2.6.2 HEC-RAS output for the lower reaches of the Mgeni river

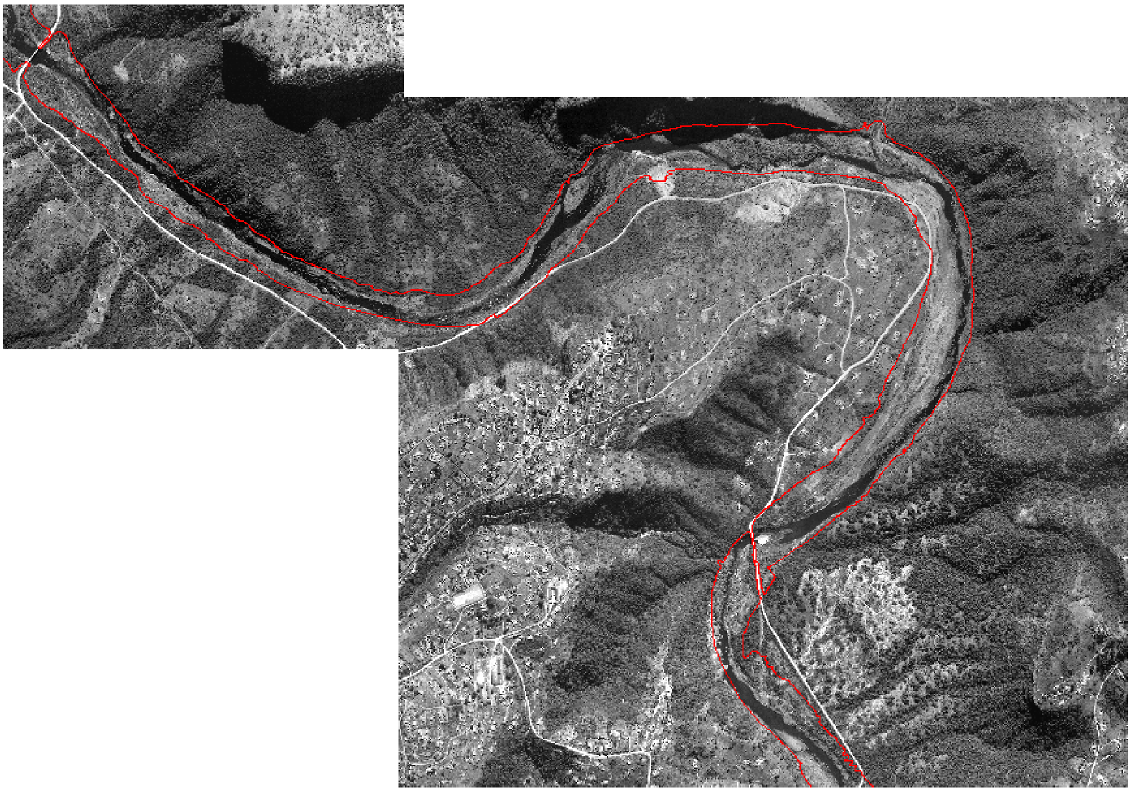


Figure 2.6.3 A 2% Annual exceedance probability (50 year Recurrence Interval) flood line: (Mgeni catchment Downstream of Inanda dam)

CHAPTER 2 THE FLOOD FORECASTING SYSTEM

SUMMARY

Several innovations to products related to flood nowcasting have been achieved in this study, marking useful technical and conceptual improvements in line with the objectives:

- Applying the Kalman Filter to a linear rainfall/runoff model to nowcast future flood flows
- Testing and choosing the "best" spatial rainfall estimation algorithm used to merge the information from rain gauges and weather radar
- Testing and choosing the "best" model to nowcast rainfields and to assess the errors
- Using GIS to automatically calculate the rainfall on sub-catchments in real time
- Devising a method and adapting a software based system to assess the rainfall and streamflow data information stream in real time
- Producing a computationally feasible algorithm to determine inundation depth from flow for the purposes of proactive Disaster Management

The concluding chapter draws together and summarizes the work presented in the report and suggests a way forward.

3.1 – WERE THE PROJECT OBJECTIVES ACHIEVED ?

Accessing Rainfall data	
Ensure the reliability of the Durban Radar.	☑
Understand and write software to handle the MDV format data.	☑
Obtain the MDV data stream from the radar in real time.	☑
Generate catchment masks.	☑
Fitting the Catchment model	
Gather relevant historical data sets.	☒
Implement Optimization procedure.	☑
Address the issue of stability from starting values.	☑
Selection of parameter sets.	☑
Forecasting and incorporation of SBM	
Set up a forecasting version of the model.	☑
Set up a conditional forecasting implementation of SBM.	☑
Incorporate SBM in parallel forecast streams.	☑
User interface.	☑
Validation and testing	
Validation on “unseen” historical data (offline).	☒
Validation and testing in real-time (online).	☒
Report Preparation	
Write final report	☑

Table 3.1.1 Synopsis of project achievements

Each of the tasks in Table 3.1.1 is discussed on an individual basis and the achievements and work programme deviations explained.

Accessing Rainfall Data:

- *Ensure the reliability of the Durban Radar*. This task was completed as outlined in Section 1.3.
- *Understand and write software to handle MDV format data*. This task was completed and is discussed briefly in Section 2.6.1. The relevant source code is discussed in Appendix B.

- *Obtain the MDV data stream directly in real-time*. Currently the transfer of data is achieved using an internet connection. The generally large size of MDV format files has prohibited direct transfer of this data from the METSYS server. The problem has been overcome by developing software to process the data at the METSYS offices and extract the relevant information which is suitably reformatted, compressed and stored on the METSYS FTP server in a two-day buffer. An automated data collection process has been developed and installed at the Umgeni Water and eThekwini Municipality offices. The METSYS FTP server is queried for new data at frequent intervals and new data downloaded to the client machine.
- *Generate catchment masks*. This task was completed as outlined in Section 1.3.

Model Fitting:

- *Gather relevant historical data sets*. This task has not been completed in full. The progress of other tasks did not allow sufficient time during the project.
- *Implement Optimisation procedure*. This work is described in Volume Two of the report.
- *Address the issue of stability*. This task was completed.
- *Selection of Parameter sets*. Parameter sets were chosen however only limited validation was carried out as described in Section 2.5.3.

Forecasting and inclusion of SBM:

- *Set up a forecasting version of the model*. This task has been completed. Software has been written to accept rainfall and streamflow inputs from text files and implement the Kalman filtering algorithm for model state updates.
- *Set up a conditional forecasting implementation of SBM*. This task has been completed. A forecasting implementation of the "String of Beads" rainfall simulation model has been developed and compared with the S_PROG model developed in Australia.
- *Incorporate SBM in parallel forecast streams*. This task has been completed with the parallelisation of the forecasts streams controlled by Python scripts and batch files.
- *User interface*. This task was originally envisioned in terms of the development of a custom front-end system for viewing the forecasting system outputs. Facilities have, instead, been developed to integrate the forecast products into a GIS as discussed in Section 2.6.

Validation and testing:

- *Validation on "unseen" historical data (offline)*. Some validation of the procedures developed has been carried out using data from the Liebenbergsvlei catchment which is of a similar size to the Mgeni catchment. Validation could not be carried out on the Mlazi catchment due to the lack of appropriate data. Data from the Mgeni catchment was to have been collected as part of another task which was not completed.
- *Validation and testing in real-time (online)*. Online testing could not be completed due to delays in obtaining the real-time rain gauge and stream gauge data streams.

At the time of writing the streamflow gauges at Inanda and Shongweni dams have been installed (August 2003) and data is being regularly published on DWAF's web page. Software has been developed to automatically retrieve this data from the web page. Testing of real-time rain gauge data retrieval has also begun at METSYS. The radar data images are collected and displayed operationally as discussed in Section 2.6.

3.2 – FINAL COMMENTS

In the year 2000, the eThekwini Municipality Disaster Management Centre did not have any facility for anticipating floods except from emergency weather reports and forecasts. They typically found themselves reacting to information phoned in by people who had either experienced damage or who had noticed that flooding was occurring. In the year 2003 they have, in the Disaster Management Centre, a GIS display overlain by real time images of rainfall measured by radar at 5-minute intervals showing them where the rain is and has fallen. They also have information coming from the Shongweni and Inanda Dams indicating what the river flows are in the two major rivers affecting their area of responsibility. These flows are converted to inundation levels routinely.

The people living near rivers have now got the potential for some warning about impending floods and the knowledge that the Disaster Management Group is working towards mitigating floods in their area in a proactive rather than reactive way. Major industrial developments have been established in Mgeni Park and in the Mlazi Basin. Some of these are strategic industries. With the flood forecasting capability in the Durban Metro Disaster Management Centre, 6 to 12 hour warning of an impending flood will enable industry to evacuate staff and perform controlled shut downs or take steps to reduce the damage to the sensitive plants. This is a far cry from September 1987 when the SAPREF Refinery and Mondi paper mill were closed for 10 days with serious economic consequences.

Much has been accomplished since the inception of this project in 2001. A key outcome has been that the beginning of a co-operative relationship between the participating organizations is becoming apparent. Where the responsibilities of separate organizations overlap partnership is an appropriate way to achieve common aims. Very little would have been achieved by the principle researcher without full co-operation and willingness to help from all parties. It is hoped that this relationship will continue to mature, as without effective, and continuing, leadership and communication the system is doomed to failure. According to du Plessis (2002) "A well-developed FFWRS, which is not implemented, is of no value.", (FFWRS - Flood Forecasting, Warning and Response System).

An example of such co-operation has been the development and implementation of a mechanism allowing the real-time transfer of data via FTP. This allowed Scott Sinclair (then of Umgeni Water - he is now engaged in Doctoral studies funded by the WRC project K5/1429) to provide a facility for visualization of the Radar data in a GIS format. The METSYS team was more than happy to install the data processing software on their server as well as providing disk space. The disaster management team provided assistance in establishing the appropriate contacts within their organization. Subsequently the Durban Metro GIS department allowed the installation of software, to collect data via FTP, on their server. The result is a display of current rainfall conditions in the Disaster management emergency response centre.

CHAPTER 3 CONCLUSIONS

In addition to the original aims of the project (all of which were achieved except those relating to real-time data hydrological data acquisition) useful work was done on rainfall forecasting and estimation. The forecasting of rainfields in real time gives good forecasts up to an hour ahead over a large area; the improved estimation comes from the merging of information from rain gauges with that from radar.

In conclusion, the pilot project has been successful enough in its application in Durban to persuade the WRC to fund a project: K5/1429: National Flood Nowcasting System: Towards an Integrated Mitigation Strategy, which incorporates personnel in SAWS:METSYS, DWAF & NDMC under the leadership Civil Engineering of the University of Natal, Durban.

THE MISP ALGORITHM (AFTER TODINI, 1978)

State Correction:

$$\nu_t = z_t - H_t \hat{x}_{t/t-1} - \bar{\nu}_t$$

$$\bar{\nu}_t = \bar{\nu}_{t-1} + \frac{\nu_t}{t}$$

$$R_{t/t} = R_{t/t-1} + \frac{\nu_t \nu_t^T - C^o_{t/t-1}}{t}$$

$$C^o_{t/t} = H_t P_{t/t-1} H_t^T + R_{t/t}$$

$$K_t = P_{t/t-1} H_t^T \left(C^o_{t/t}\right)^{-1}$$

$$\hat{x}_{t/t} = \hat{x}_{t/t-1} + K_t \nu_t$$

$$\bar{w}_t = \bar{w}_{t-1} - \frac{\left(\Gamma_t^T \Gamma_t\right)^{-1} \Gamma_t^T K_t \nu_t}{t}$$

$$Q_{t/t} = Q_{t/t-1} + \frac{\left(\Gamma_t^T \Gamma_t\right)^{-1} \Gamma_t^T K_t \left(\nu_t \nu_t^T - C^o_{t/t-1}\right) K_t^T \Gamma_t \left(\Gamma_t^T \Gamma_t\right)^{-1}}{t}$$

State Prediction:

$$\hat{x}_{t+1/t} = \phi_{t+1/t} \hat{x}_{t/t} + \Gamma_{t+1} \bar{w}_t + \beta u_t$$

$$P_{t+1/t} = \phi_{t+1/t} P_{t/t} \phi^T_{t+1/t} + \Gamma_{t+1} Q_{t/t} \Gamma^T_{t+1}$$

$$C^o_{t+1/t} = C^o_{t/t}$$

$$R_{t+1/t} = R_{t/t}$$

$$Q_{t+1/t} = Q_{t/t}$$

Parameter Filter:

$$\nu_t^* = H_t K_t \nu_t$$

$$R_t^* = H_t K_t C^o_{t/t} K_t^T H_t^T$$

$$H_t^* = H_t F_t$$

$$C_t^{o*} = H_t^* P^*_{t/t-1} H_t^{*T} + R_t^*$$

$$K_t^* = P^*_{t/t-1} H_t^{*T} \left(C_t^{o*}\right)^{-1}$$

$$\hat{\theta}_{t+1/t} = \hat{\theta}_{t/t-1} + K_t^* \nu_t^* + \Gamma^*_{t+1} \bar{w}^*$$

$$P^*_{t+1/t} = \left(I - K_t^* H_t^*\right) P^*_{t/t-1} \left(I - K_t^* H_t^*\right)^T + K_t^* R^*_{t/t} K_t^{*T} + \Gamma^*_{t+1} Q^* \Gamma^{*T}_{t+1}$$

where the subscripts on the terms indicate the time step for which the quantity has been estimated and how much information is used in the estimate. For example,

> $\hat{x}_{t+1/t}$ is the *a priori* forecast of the system state at time t+1, given information up to time t.

The terms in the equations are:

$\hat{x}$	is an estimate of the system state vector [n, 1].
z	is the observed system measurement vector [m, 1].
ϕ	is the state transition matrix [n, n].
$\overline{v}$	is the mean measurement error [m, 1].
$\overline{w}$	is a vector of errors in the model schematisation [mw, 1].
H	is the state to measurement transformation matrix [m, n].
Γ	is a dimensional conversion matrix [n, mw].
Q	is the covariance matrix of model schematisation errors [mw, mw].
R	is the measurement error covariance matrix [m, m].
ν	is the innovation of the filter [m, 1].
C^o	is the normalization matrix [m, m].
K	is the Kalman gain matrix [n, m].
P	is an estimate of the system state error covariance matrix [n, n].
u	is a vector of exogenous inputs [ex, 1].
β	is a dimensional transformation matrix from inputs to states [n, ex].
$\hat{\theta}$	is the parameter vector [p, 1].
ν^*	is the innovation of the parameter filter [m, 1].
R^*	is the parameter error covariance matrix [m, m].
F	is a projection matrix [n, p].
H^*	is the parameter to measurement transfer matrix [m, p].
C^{o*}	is the parameter normalisation matrix [m, m].
P^*	is the parameter error covariance matrix [p, p].
K^*	is the gain of the parameter filter [p, m].
$\overline{w}^*$	is the parameter system noise mean [ms, 1].
I	is an identity matrix [p, p].
Q^*	is the parameter system noise covariance [ms, ms].
Γ^*	is a dimensional conversion matrix for the parameters [p, ms].

SOFTWARE

The source code has been written (as far as possible) in standard C++ (ISO, 1998) and has been shown to compile with Microsoft's Visual C++ 6.0 compiler. In many cases the code has also been compiled with GCC-3.2 (GNU, 2002) as a check. C++ was chosen as the programming language of choice due to Scott Sinclair's familiarity with the language and it's predecessor the C programming language. There are many more important reasons for choosing this language for the development of computationally intensive scientific code. C++ is a high level Object Oriented (OO) language allowing the abstraction of mathematical concepts away from the manipulation of raw memory into a form more familiar to the scientist or engineer. Mathematical constructs may then be written in the language of the problem rather than (as in FORTRAN for example) the language of the machine. More than just providing an OO framework C++ is able to produce highly efficient code with performance comparable to and in some cases better than the equivalent FORTRAN code (Veldhuizen, 1997). A further advantage of the C++ language is its portability among different machine architectures and operating systems. This was a particularly attractive feature, because the computer operating system used by METSYS is Linux while the development of software to process radar data was to be carried out in a Microsoft Windows environment at Umgeni Water.

Some of the software products described have also been written in the Python language (Python, 2003). Python differs from C++ not only in its syntax but more fundamentally because it is an interpreted language while C++ is a compiled language. The advantage of an interpreted language is that commands can be executed as soon as they are entered and this can speed up the design and testing of the system. The disadvantage is that the interpreter (software which literally interprets the commands) creates a barrier between the code and the machine resulting in severe performance reductions. Python was therefore used where performance depended on outside factors, such as the FTP downloads where the time spent transferring information between machines over the internet was several orders of magnitude greater than the time taken to request the information.

The software developed during the course of this project is copyright to Scott Sinclair and the Water Research Commission. The details of the copyright are stated in each C++ source file in a header, an example of which is shown in figure B1.

```
/*************************************************************************

mispx.cpp

Written: 13/05/2002 by Scott Sinclair

Last Modified: 07/01/2003 by Scott Sinclair

Copyright (c) 2002 Scott Sinclair and Water Research Commission.

Permission to copy, use, modify, sell and distribute this software
is granted provided this copyright notice appears in all copies.
This software is provided "as is" without express or implied
warranty, and with no claim as to its suitability for any purpose.

*************************************************************************/
```

Figure B1 Standard copyright notice

The source code is available from the University of Natal. Enquiries should be directed to Professor Geoff Pegram, Civil Engineering Programme, University of Natal, 4041, Durban, South Africa.

B1 SOFTWARE DOCUMENTATION

Documentation of the C++ software has been produced using a freely available program called Doxygen (van Heesch, 2002). Doxygen is a source code documentation system which allows the source code and appropriate comments on the use of all functions and classes to be extracted directly from the source code to a set of HTML documents. The documents are hyper linked allowing easy viewing of the documentation using any standard web browser. The documentation could even be used to form a portion of a website. Figure B2 shows a screen capture of the documentation system.

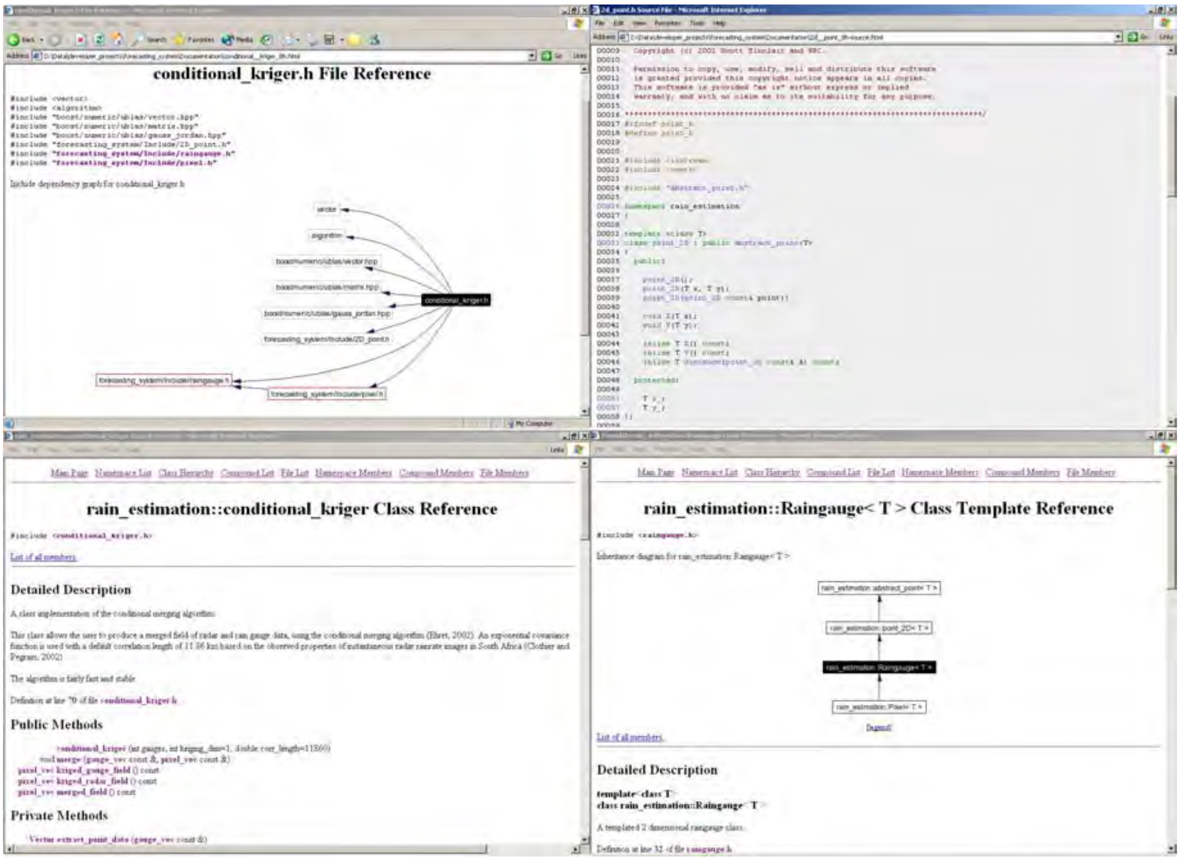


Figure B2 Screen captures showing the Doxygen documentation system

The documentation can be easily updated to reflect any changes in the source code which means that it is easy to maintain the documentation as the software evolves.

B2 RADAR AND RAIN GAUGE MERGING SOFTWARE

Software was written to merge radar and rain gauge estimates of rainfall using the methods described in Section 2.2. An implementation of the conditional merging technique (Ehret, 2002) was developed to accept as input radar data in the MDV format and the appropriate set of rain gauge readings in a text format. The output is an MDV file containing the resulting merged field. Figure B3 shows an example of the text format required for the rain gauge inputs. The first two numbers (in brackets) indicate the X and Y co-ordinates (in metres) of the gauge in a Transverse Mercator projection. The third number is the accumulated rainfall for the period considered in millimetres.

```
(-3217.92, -3.31431e+006)     0
(-3111.94, -3.30976e+006)     35
( 563.228, -3.3125e+006)      22
( 992.658, -3.30911e+006)     27
(-8320.69, -3.304e+006)       31
(-3462.41, -3.30421e+006)     36
( 1100.64, -3.30233e+006)     0
(-25112.4, -3.29688e+006)     24
(-16708.8, -3.29487e+006)     29.657
```

Figure B3 Text format for the rain gauge inputs

B3 KALMAN FILTERING SOFTWARE

The Kalman filtering software developed for this project is a full implementation of the MISP filter (Todini, 1978) which is specialised to the Kalman filter when no parameter updates are required. The input to the program is a parameter file which sets up the model structure and gives the input and output data file locations. A small dimension example parameter file used in Section 2.3.2 follows:

```
# A parameter file for the MISP Algorithm

# Path to the file containing system measurements
MEASUREMENT_FILE  D:\developer\misp\observed.txt

# Path to the file containing system inputs
INPUT_FILE  D:\developer\misp\rain.txt

# Path to the output file
OUTPUTS     D:\developer\misp\forecast.txt

# Frequency of parameter updates. No updates if this is equal to -1
FREQUENCY   -1

# Matrix sizes
STATES                 3     (n)
MEASUREMENTS           1     (m)
STATE_NOISE            3     (mw)
PARAMETERS             7     (p)
PARAMETER_NOISE        7     (ms)
INPUT_SIZE             1     (ex)

# Initial parameter vector (of variable parameters). [px1]
# The parameters here are k[i]'s of the linear reservoir model
PARAM_VECTOR
21    15    11    22    0.1   0.1   0.1

# Initial states, vector [nx1]
START_STATES
0     0     0

# Exogenous input conversion, matrix [nxex]
```

```
BETA
1
0
0

# System noise conversion, matrix [nxmw]
SYS_GAMMA
1     0     0
0     1     0
0     0     1

# System noise mean, vector [mwx1]
SYS_NOISE_MEAN
0     0     0

# System noise covariance, matrix [mwxmw]
SYS_NOISE_COVARIANCE
1     0     0
0     1     0
0     0     1


# System error covariance, matrix [nxn]
SYS_ERROR_COVARIANCE
10000 0     0
0     10000 0
0     0     10000

# Parameter to measurement transfer conversion, matrix [pxn]
C
0     0     0
0     0     0
0     0     0
0     0     1
0     0     0
0     0     0
0     0     0

# Measurement noise mean, vector [mx1]
MEA_NOISE_MEAN
0

# Measurement noise covariance, matrix [mxm]. Should be non-singular.
MEA_NOISE_COVARIANCE
1000

# Parameters system noise mean, vector [msx1]
PAR_SYS_NOISE_MEAN
0
0
0
0
0
0
0

# Parameters system noise covariance, matrix [msxms]
PAR_SYS_NOISE_COVARIANCE
1e-5  0     0     0     0     0     0
0     1e-5  0     0     0     0     0
0     0     1e-5  0     0     0     0
0     0     0     1e-5  0     0     0
0     0     0     0     1e-5  0     0
0     0     0     0     0     1e-5  0
0     0     0     0     0     0     1e-5
```

```
# Parameters system noise conversion, matrix [pxms]
PAR_GAMMA
1     0     0     0     0     0     0
0     1     0     0     0     0     0
0     0     1     0     0     0     0
0     0     0     1     0     0     0
0     0     0     0     1     0     0
0     0     0     0     0     1     0
0     0     0     0     0     0     1

# Parameter system error covariance, matrix [pxp]
PAR_SYS_ERROR_COVARIANCE
10    0     0     0     0     0     0
0     10    0     0     0     0     0
0     0     10    0     0     0     0
0     0     0     10    0     0     0
0     0     0     0     10    0     0
0     0     0     0     0     10    0
0     0     0     0     0     0     10


# Matrix of constant parameters in the state transition matrix, [nxn]
A
1     0     0
0     1     0
0     0     1

# Fixed members of Parameter to measurement transfer conversion,
# matrix [mxn]
D
0     0     0

# 3-Dimensional array for transition of parameters
# from p-space to n-space, [nxnxp] This defines the model structure.
B_3D
-1    -1    0     0     -1    0     0
0     0     0     0     0     0     0
0     0     0     0     0     0     0

1     0     0     0     0     0     0
0     0     -1    0     0     -1    0
0     0     0     0     0     0     0

0     1     0     0     0     0     0
0     0     1     0     0     0     0
0     0     0     -1    0     0     -1
```

B4 RADAR TO GIS CONVERSION SOFTWARE

This software was written in C. The executable, which runs routinely on the METSYS server in Bethlehem, accepts an MDV file as input and produces a geo-referenced bitmap image as output. The bitmap image can then be viewed as a theme in the Arcview GIS. Arcview scripts have also been developed to facilitate the process of selecting and updating the images.

B5 MDV MASKING AND ACCUMULATION SOFTWARE

The software accepts as input a mask file representing the catchment area to be masked. This file is in MDV format. The other command line arguments are the accumulation period in seconds and an optional search directory (the default is the current directory). The software processes all valid MDV files in the search directory that contain data within the time-period bounded by the current system time and the current time less the accumulation period. The accumulation for the masked region is produced in a text format.

B6 FTP DOWNLOAD SOFTWARE

A Python script was developed to query a remote FTP server and download the contents of a particular directory to the client machine. This runs routinely at the eThekwini Municipality offices to update the GIS compatible radar images and is controlled by a simple batch file and the Microsoft Windows task scheduler.

B7 WEB PAGE QUERY SOFTWARE

A Python script has been written which connects to the DWAF web page and retrieves the most recent values of flow at the outlets of the Inanda and Shongweni dams. This script is not running at eThekwini Municipality (at the time of writing, September 2003) but is running at the University of Natal on a test machine.

Bab-Hadiashar A., Suter D. and Jarvis R., 2-D Motion extraction using an image interpolation technique, *SPIE*, 2564, 271-281, 1996.

Barron G.W., HEC-RAS, River Analysis System Hydraulic Reference Manual, *US Army Corps of Engineers, Hydrologic Engineering Center*, 2001.

Brunner E.J., Duffy C., Frakes B., Lakhtakia M.N., Miller D.A, Schwartz F.W., White R.A., Yarnal M.N. and Yu Z., Simulating the river-basin response to atmospheric forcing by linking a mesoscale meteorological model and hydrologic model system, *Journal of Hydrology*, 218, 72-91, 1999.

Camus T. and Bulthoff H.H., Real-time optical flow extended in time, *Max-Planck-Institut*, Technical report No. 13, 1995.

Clothier A.N. and Pegram G.G.S., Space-time modeling of Rainfall using the String of Beads model: Integration of Radar and Raingauge data, *Water Research Commission Report*, No. 1010/1/02, 2002.

Cowpertwait P.S.P, Further developments of the Neyman-Scott clustered point process for modeling rainfall, *Water Resources Research*, 27(7), 1431-1438, 1991.

Ehret U., Rainfall and flood nowcasting in small catchments using weather radar, *PhD Thesis*, University of Stuttgart, 2002.

Engelbrecht F.A., deW Rautenbach C.J., McGregor J.L. and Katzfey J.J., January and July climate simulations over the SADC region using the limited-area DARLAM, *Water SA*, 28(2), 361-374, 2002.

Gelb A., Applied Optimal Estimation, *MIT Press*, Cambridge, MA, 1974.

Grecu M., Krajewski W.F., A large-sample investigation of statistical procedures for radar-based short-term quantitative precipitation forecasting, *Journal of Hydrology*, 239, 69-84, 2000.

GNU, The GNU compiler collection, *gcc.gnu.org*, 2002.

Hannesen R., Quantitative precipitation estimation from radar data – A review of current methodologies, *Progress report for MUSIC*, European Commission contract No. EVK1-CT-2000-00058, 2002a.

Hannesen R., An enhanced surface rainfall algorithm for Radar Data, *Progress report for MUSIC*, European Commission contract No. EVK1-CT-2000-00058, 2002b.

ISO, Programming languages --- C++, *International ISO/IEC Standard 14882*, 1998.

Kalman R.E., A new approach to linear filtering and prediction problems, *Transaction of the ASME – Journal of Basic Engineering*, 35-45, 1960.

Lardet P. and Obled C., Real-time flood forecasting using a stochastic rainfall generator, *Journal of Hydrology*, 162, 391-408, 1994.

Mittermaier M.P., Terblanche D.E., Converting weather radar data into Cartesian space: A new approach using DISPLACE averaging, *Water SA*, 43(1), 46-50, 1997.

Pegram G.G.S., Spatial interpolation and mapping of rainfall: 3. Optimal integration of rain gauge, radar & satellite-derived data in the production of daily rainfall maps, *Progress report to the Water Research Commission*, for the period April 2001 to March 2002 on contract K5/1153, April 2002.

Pegram G.G.S., Clothier A.N., High Resolution Space-Time Modelling of Rainfall: The "String of Beads" Model, *Journal of Hydrology*, 241, 26-41, 2001.

Pegram G.G.S., and Sinclair D.S., A linear catchment model for real time flood forecasting, *Water Research Commission Report*, No. 1005/1/02, 2002.

Press W.H., Teukolsky S.A., Vetterling W.T., Flannery B.P., Numerical Recipes in C – The art of scientific computing, *Second Edition*, *Cambridge University Press*, 1992.

Python, The Python language website, *www.python.org*, 2002.

Seed A.W., Srikanthan R. and Menabde M., A space and time model for design storm rainfall, *Journal of Geophysical Research*, 100(D21), 31623-31630, 1999

Seed A.W., A dynamic and spatial scaling approach to advection forecasting, *Proceedings of the Fifth International Symposium on Hydrological Applications of Weather Radar – Radar Hydrology*, Kyoto - Japan, 2001

Sinclair D.S., Ehret U., Bardossy A. and Pegram G.G.S. (2003). Comparison of Conditional and Bayesian Methods of Merging Radar & Raingauge Estimates of Rainfields, *Oral Presentation EGS/AGU/EUG Joint Assembly*, Nice, France, April.

Todini E., Mutually Interactive State/Parameter Estimation (MISP) – Application of Kalman Filter to Hydrology, *Hydraulics and Water Resources*, Chao-Lin Chiu Ed., University of Pittsburg, Penn., 1978.

Todini E., A Bayesian technique for conditioning radar precipitation estimates to rain-gauge measurements, *Hydrology and Earth System Sciences*, 5(2), 187-199, 2001.

Van Heesch D., Doxygen a documentation system for C++, C, Java, IDL (Corba, Microsoft, and KDE-DCOP flavors) and to some extend PHP, *www.doxygen.org*, 2002

Veldhuizen T.L., Jernigan M.E., Will C++ be faster than FORTRAN ?, *Proceedings of the 1st International Scientific Computing in Object Oriented Parallel Environments (ISCOPE'97)*, 1997

Wilson J.W., Brandes E.A., Radar measurement of rainfall – A summary, *Bulletin of the American Meteorological Society*, 60(9), 1048-1058, 1979

WPDM, White paper on disaster management, *Department of constitutional development*, South Africa, 1998

Young P., Recursive estimation and Time series analysis, *Springer-Verlag*, Berlin, 1984.