REAL-TIME FLOW FORECASTING

By

HAMED ASSAF

B. Eng. (Civil Engineering) Kuwait University

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES

CIVIL ENGINEERING

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

June 1991

© HAMED ASSAF, 1991
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Civil Engineering

The University of British Columbia
Vancouver, Canada

Date June 28, 1991
Abstract

The main objective of this research is to develop techniques for updating deterministic river flow forecasts using feedback of real-time (on-line) flow and snowpack data. To meet this objective, previous updating methods have been reviewed and evaluated and typical error patterns in flow forecasts have been analyzed using standard techniques. In addition, a new criterion based on the coefficient of determination and coefficient of efficiency has been introduced to evaluate systematic errors in flow forecasts. Moreover, lagged linear regression has been suggested as a method for detecting and estimating timing errors.

Arising from this initial work, two different updating procedures have been developed. Further work has shown that these two independent procedures can be usefully combined, leading to yet further improvement of forecast. Arising from these methods, two other additional approaches have been formulated, one for correcting timing errors and one for updating snowpack estimation parameters from flow measurements.

The first of the updating methods consists of a flow updating model which was developed to update the flow forecasts of the UBC watershed model using the most recent flow measurement. The updating process is achieved using the Kalman filter technique. The performance of the updating model is mainly controlled by the relative values of two parameters of the Kalman filter technique: the measurement variance and the state variance. It is found that the measurement variance is best selected as the square of a percentage of the flow. The updating model has been applied on the Illecillewaet river basin in British Columbia. A significant improvement in flow forecasts has been observed.

The second method has been developed to update parameters of an energy budget
snowpack model using on-line snowpack measurements. The updating procedure is based on calculating the value of a snowpack parameter that yields a perfect correspondence between measured and calculated snowpacks. The updated value is then used in the snowpack model to enhance its future forecasts with feedback from previous snowpack measurements. The snowmelts generated by the updated snowpack model are then routed to produce flow forecasts. Applying this model on the snowpack measured at Mt. Fidelity in the upper Columbia River Basin in British Columbia showed that both the snowpack forecasts and the flow forecasts generated from these updated snowpack forecasts were greatly improved.

Because the above two updating methods operate independently, they can be applied in combination whenever an appropriate measurement is available. The combined use of these two methods to data from the Illecillewaet river basin showed an additional improvement in flow forecasts.

As a further development, the snowpack estimation model has been adapted so that a Kalman filter approach can be used to update snowpack estimation parameters from flow measurements.

It is finally concluded that flow forecast updating requires the application of several methods, rather than one simple approach, because errors arise from various sources. In addition, updating procedures may prove useful in achieving a better calibration for watershed models.
# Table of Contents

Abstract .......................................................... ii

List of Tables ....................................................... ix

List of Figures ...................................................... x

Acknowledgement .................................................... xiv

1 INTRODUCTION ...................................................... 1

2 Analysis of Errors in Flow Forecasts ......................... 4
   2.1 Sources of Errors ............................................. 4
       2.1.1 Model Structure ......................................... 5
       2.1.2 Model Parameters ........................................ 7
       2.1.3 Input Data ................................................ 8
       2.1.4 Reference Data Error .................................... 9
   2.2 Error Analysis Methods ...................................... 10
       2.2.1 Graphical Presentations ................................ 10
       2.2.2 Least Squares Criteria .................................. 13
       2.2.3 The Autocorrelation and the Partial Autocorrelation Functions 22
   2.3 Summary and Conclusion ..................................... 26

3 Approaches to Real-Time Flow Forecasting ................... 28
   3.1 Error Prediction Models ..................................... 29
5.6.1 Overall Performance .............................................. 77
5.6.2 Examining Progressive Model Performance ................. 83
5.6.3 Detection of Linearity in The Flow Forecasts Residuals .. 91
5.6.4 Summary and Conclusion ......................................... 92

6 A Snowpack Updating Model ....................................... 94
6.1 The Energy Budget of the Snowpack ............................. 96
6.1.1 The Shortwave Heat Influx ...................................... 97
6.1.2 The Longwave Radiation ......................................... 97
6.1.3 The Convective and Adective Heat Transfer ................. 98
6.1.4 The Rain Melt ................................................... 98
6.2 Calculating the Snowpack in the UBC Watershed Model ... 98
6.3 Sensitivity Analysis of the Snowpack Parameters .......... 101
6.3.1 The Cloud Cover ................................................ 101
6.3.2 The Snow Albedo ............................................... 107
6.3.3 The Snow Formation Function ................................. 109
6.4 The Snowpack Updating Approach ............................... 110
6.5 The Snowpack as a Function of the Net Energy Fluxes and the Snowfalls 112
6.6 The Snowpack Parameters As a Function of the Snowpack ... 114
6.6.1 The Cloud Cover Correction Factor $\eta_c$ ................. 114
6.6.2 The Cloud Cover Parameter $\Delta T_s$ ....................... 116
6.6.3 The Snow Albedo Correction Factor $\eta_a$ ................. 116
6.6.4 The Snow Formation Parameter $T_r$ ......................... 117
6.7 Algorithm of the Snowpack Updating Model .................. 118
6.8 Application of the Snowpack Updating Model ............... 119
6.9 Combined Application of the Snowpack and the Flow updating Models 123
6.10 Updating Parameters of the Snowpack Using Flow Measurements . . . . 131
6.11 Summary and Conclusion ........................................... 131

7 Discussion and Conclusions ........................................ 134
  7.1 Sources and Analysis of Forecast Residuals ................. 135
  7.2 Literature Review of the Real-Time Forecasting Models ....... 137
  7.3 The Kalman Filter .................................................. 140
  7.4 The Proposed Updating Procedures .............................. 143
    7.4.1 Direct Updating of the Flow Forecasts ................. 144
    7.4.2 A Snowpack Updating Model ............................... 146
  7.5 Updating Snowpack Parameters Using Flow measurements ....... 149
  7.6 Detection and Correction of Timing Errors ................. 149
  7.7 Combined Use of Updating Techniques ......................... 150
  7.8 General ............................................................ 150

REFERENCES .............................................................. 153

APPENDICES ............................................................... 159

A The Illecillewaet Watershed ......................................... 159

B Detailed Algorithm of the Flow Updating Model ................. 163
  B.1 Prediction .......................................................... 163
    B.1.1 The Initial Conditions .................................... 163
    B.1.2 For any time t > 0.0 ....................................... 164
  B.2 Updating .......................................................... 164
    B.2.1 The Kalman Gain ........................................... 164
    B.2.2 The Weighting Factors .................................... 165
B.2.3 The Covariance Matrix ................................. 166

B.3 A Sample Calculation ...................................... 167
  B.3.1 Prediction ............................................. 167
  B.3.2 Updating .............................................. 168
  B.3.3 Prediction for the Following Day ..................... 171

C Derivation of The Snowpack Function ..................... 173

D Derivation of the Flow-Snowpack Updating Model ......... 181
  D.1 Development of the Basic Model ....................... 181
    D.1.1 Updating $\eta_c$ .................................. 183
    D.1.2 Updating $\Delta T_s$ .............................. 183
    D.1.3 Updating $\eta_a$ .................................. 184
List of Tables

5.1 Evaluation of the flow forecasts in terms of several criteria. ........... 82
List of Figures

2.1a Observed flows and their estimates and residuals from a misrepresentative model. ................................................. 6
2.1b Observed flow vs. forecast from a misrepresentative model. .......... 7
2.2 Examples of plots of observed, forecast flows, residuals, and accumulated errors. .............................................................. 11
2.3a Example of a linear plot of observed flow vs. forecast. ............... 12
2.3b Example of a logarithmic plot of observed flow vs. forecast. .......... 12
2.4 Example of a plot of residual vs. residual one day before. ............. 13
2.5 Hydrographs of the observed, the UBC and the timing error-updated flows and their residuals for a forecast lead of five days. ............... 21
2.6 Theoretical autocorrelation function of $AR(1)$. ........................ 24
2.7 Theoretical autocorrelation function of $AR(2)$. ........................ 24
2.8 Example of an autocorrelation function of flow forecast residuals. ... 25
2.9 Example of a partial autocorrelation function of flow forecast residuals. 25
3.1 Flow chart of a basic conceptual model. ................................. 36
4.1a Effect of a good measurement and a poorly structured state equation on updating ......................................................... 48
4.1b Effect of a bad measurement and a well-structured state equation on updating ............................................................. 48
4.2 The Kalman filter as a Bayesian approach. .............................. 49
4.3 Actual and measured flows and measurement errors (simulated data). . 51
5.10c The daily estimates of the weighting factors for percentage-measurement variance of 15%.

5.11 Successive daily updating of flow forecasts for forecast leads of 1, 2, ..10 days \((R = (15\% \text{flow})^2)\).

5.12 The key figure for Figs. 5.9 and 5.11.

6.1 Flow chart of the UBC snowpack model.

6.2 Sensitivity of the snowpack to the cloud cover parameter \(\eta_c\).

6.3 The effect of under- or over-estimating the cloud cover on the snowpack.

6.4 Sensitivity of the snowpack to the cloud cover parameter \(\Delta T_c\).

6.5 Sensitivity of the snowpack to the snow albedo parameter \(\eta_a\).

6.6 Sensitivity of the snowpack to the snow formation parameter \(T_r\).


6.8 The updated values of the cloud cover parameter \(\eta_c\) for 1969-1970.

6.9 Hydrographs of the observed, un-updated and updated flow forecasts and their residuals for 1970.


6.11 The updated values of the cloud cover parameter, \(\eta_c\) for 1974-1976.

6.12 Hydrographs of the observed, un-updated and updated flow forecasts and their residuals for 1975.

6.13 Hydrographs of the observed, un-updated and updated flow forecasts and their residuals for 1976.

6.14 Hydrographs of the observed, updated (snowpack) and updated (snowpack and flow) flow forecasts and their residuals for 1970.

6.15 Hydrographs of the observed, updated (snowpack) and updated (snowpack and flow) flow forecasts and their residuals for 1975.
6.16 Hydrographs of the observed, updated (snowpack) and updated (snowpack and flow) flow forecasts and their residuals for 1976 . . . . . . . . . . . . . . 130
Acknowledgement

Great thanks and appreciation to my parents, Shafe and Fiaza, for their undying support throughout these long years in school.

I greatly appreciate the support and advice of my research supervisor, Dr. Michael Quick throughout my research and in preparing this thesis. I also thank the supervisory committee members: Dr. Denis Russell, Dr. Bill Caselton and Dr. Al Freeze for their valuable comments on my thesis.

Special thanks to my wife Emily Mulleda for her support and help in preparing this thesis.

This research was funded for the last two years by the B.C. Relief Fund Administated by the Vancouver Foundation. This funding supports ongoing research in the Department of Civil Engineering on flood problems of importance to British Columbia.

This research was also funded by a graduate fellowship awarded by the University of British Columbia and by Earl R. Peterson Memorial Fellowship.
Chapter 1

INTRODUCTION

Operation and management of a watershed system requires accurate and reliable flow forecasts. The accuracy of short-term flow forecasts is particularly important for effective flood control measures. Traditionally, flow forecasts were generated from physically-based models that simulate the behavior of the watershed system based on "presumably" error-free input of meteorological data (precipitation and temperature). The flow measurements were only used as reference data for calibrating model parameters. However, advances in processing hydrological data made it possible to have fast and reliable access to flow measurements. This encouraged hydrologists to develop real-time forecasting models, i.e. models that use real-time (on-line) hydrological data, as well as hydrometeorological data, in calculating estimates of the future flows. The first generation of these models were based only on simple input-output relationship between meteorological inputs and measured flows with no consideration of the underlying physical processes. Although these models do well for very-short forecast leads, their lack of structure makes them unreliable for longer forecast leads. Later, more emphasis was placed on developing physically-based real-time forecasting models. The characteristics of both types of models with examples from the literature are discussed in Chap. 3.

The main objective of this research is to develop techniques for incorporating feedback of real-time data (flow, snowpack) in upgrading non-real-time conceptual watershed models without altering their main structures. This approach is important since most conceptual watershed models are not designed to operate in the real-time mode and have
highly complex nonlinear structures that are difficult to adjust to operate in this mode. The UBC watershed model developed by M. C. Quick and A. Pipes (1989b) is a typical example of these models. It has been selected as a study object for this research. Study data are selected from the Illecillewaet basin in British Columbia.

Since the main objective of the updating procedure of any flow forecast is to reduce the residuals, or errors, in these forecasts, it became necessary to study the characteristics of these errors and review existing evaluation techniques for flow forecasts. These issues and the introduction of a new least squares criterion for detecting linear errors in flow forecasts will be presented and discussed in Chap. 2.

Reviewing the updating procedures used in many of the real-time flow forecasting models showed that most of these procedures were based on the Kalman filter technique. The theoretical and practical aspects of this technique will be presented and discussed in Chap. 4.

The Kalman filter technique was used to develop a procedure for updating the flow forecasts from the UBC watershed model using real-time flow measurements. The updating procedure was designed based on the assumption that the flow is equal to a weighted sum of the flow components calculated by the UBC watershed model. The selection of the updating parameters and the application of this updating method to data from the Illecillewaet basin in British Columbia will be presented and discussed in Chap. 5.

The extreme flows in the mountainous watershed systems, of which the Illecillewaet basin is an example, are mainly generated by snowmelt during the warm spring and summer seasons. Snowmelt forecasting is therefore different from rainfall-induced flow forecasting, because most of the precipitation input and the snowpack are already known at the start of the season and can be updated during the melt season. Forecasts can therefore be made using historical records of temperature and precipitation for previous years, so that, by using a range of such historical records, it is possible to estimate a range
of possible flows for the whole forecast snowmelt season, say from April to September. To achieve these longer term forecasts, it is very important to have a well calibrated model and, in addition, being able to update the flow from the latest flow and snowpack information is likely to be very useful. For this reason and because of the long residence time of the snow input, the quality of long-term forecasts of high flows are expected to be sensitive to the estimation of snowmelt in the watershed model. Based on this conclusion, a method has been designed to update parameters of the snowpack routine of the UBC watershed model using real-time snowpack measurements. This method was also applied to update the cloud cover estimates in the Mount Fidelity station located on the upper part of the Illecillewaet basin. The updated cloud cover estimates were then used in the UBC watershed model to calculate flow forecasts for the Illecillewaet basin. Details of the development of this method, its algorithm and the results from its application will be presented and discussed in Chap. 6.

Due to the interaction between the above two updating techniques, the combined use of both methods was studied through combined application on data from the Illecillewaet basin.
Chapter 2

Analysis of Errors in Flow Forecasts

The purpose of any flow forecasting model is to simulate a given watershed system so that the flow at the basin outlet can be estimated from input data on precipitation and temperature. These models range from comprehensive watershed models to input-output black box models. For many reasons, to be discussed later, flow forecasts can never be perfect. The quantity and quality of such imperfections or errors vary considerably from one situation to another and can have very substantial implications on the management and the design of water resources projects and flood control measures. Studying the characteristics of these errors is essential for developing updating techniques for flow forecasts and for improving the designs and calibrations of flow forecasting models.

This chapter starts with a discussion on the main sources of these errors. It follows with a discussion of methods used in analyzing model performance and forecast errors.

2.1 Sources of Errors

Residuals in flow forecasts can be attributed to errors in the structure of the flow forecasting model, errors in the model parameters, and errors in the input and the reference data. These sources of errors are discussed in the following sections.
2.1.1 Model Structure

Watershed systems are highly complex and diverse and much is unknown about the underlying physical processes. This makes it necessary to use approximate lumped representations of these systems. Stochastic representations are used to simulate unknown processes. Time and budget constraints on data collection and operation further limit the ability of a flow forecasting model in simulating watershed behavior. However, flow forecasts may be substantially improved by applying some modifications to model structure.

Model structure plays an important part in the reliability of flow forecasts. Conceptual models, which are based on a physical representation of watershed systems, are observed to be superior to black box models, which are based on input-output relations, especially for long forecast leads (see Kitanidis & Bras, 1980c; Moore, 1986; and Georgakakos, 1986b.)

Flow forecast errors caused by a misrepresentative model are usually large and do not follow identifiable patterns (see the example shown in Fig. 2.1a.) The adequacy of a model structure can be simply tested by making linear regression between the observed hydrograph and the calculated hydrograph. A very small coefficient of determination $D$, to be defined later in section 2.2.2 (see Fig. 2.1b) means that the two hydrographs are only weakly correlated, thus indicating a serious problem in model structure.

Model over-parameterization or under-parameterization are problems that can vary dramatically in their complexities. In general, a well experienced model designer can detect and correct these problems. A simple example of model under-parameterization is if the fast flow component in a watershed model is missing resulting in underestimation of the flow peaks.
Fig. 2.1a Observed flows and their estimates and residuals from a misrepresentative model.
Fig. 2.1b Observed flow vs. forecast from a misrepresentative model.

2.1.2 Model Parameters

Model parameters can be poorly calibrated or specified for one or more of the following reasons:

1. Errors in the model structure.

2. Inadequacy of the input data for the following reasons:

   - Measurement errors.
   - Inability of the input data to excite all the watershed system modes (i.e., its responses to the possible conditions in the river basin). For example, the precipitation data collected during a dry season is useless in calibrating the fast rainfall-runoff unit hydrograph since most of the rain infiltrates into the soil and into the groundwater system.
Chapter 2. Analysis of Errors in Flow Forecasts

- Insufficient data record length. Model parameters calibrated from short data record are reliable. On the other hand, long data records provide the opportunity to detect the effects of systematic errors in the data.

- Insufficient number and poor distribution of data collection stations.

Troutman (1982) noted that if erroneous input data is used for calibrating model parameters, input data with the same type of error produces better results for the forecasting period than the ones produced by more accurate data.

3. Parameter correlation: Highly correlated parameters are difficult to calibrate, since a change in one parameter affects the other parameter(s). This is known as the identifiability problem.

4. Reference data errors: which are mainly caused by insufficient data record, measurement errors and using erroneous runoff-stage rating curves.

2.1.3 Input Data

Input data may undermine the quality of flow forecasts for one or more of the following reasons:

- Inferior measurement quality due to human and/or equipment errors;

- Misrepresentative data stations; and

- Poorly distributed data stations.

If there are consistent input measurement errors, the result will be biased flow forecasts. If these errors are reasonably linear, these errors may be reduced using a method for updating flow forecasts based on the Kalman filter technique. An example is the flow updating model presented in Chap. 5.
Misrepresentative data stations may result in very poor flow forecasts with inconsistent errors similar to those due to misrepresentative model structure (see Fig 2.1). These errors are difficult, if not impossible, to predict, so, it may be worthwhile disregarding these stations and use other sources of input data.

The effect of the density and number of precipitation stations on flow forecasts depends mainly on the time-distribution of the precipitation in the watershed, the topographical and the hydrological characteristics of the watershed, the variability and the time scale of flows, and the criterion used in evaluating the flow forecasts. A relatively uniformly distributed precipitation may be sufficiently estimated by one precipitation station, while for some events the average precipitation can be in serious error. For example, thunderstorm precipitation estimated from a single station may be either underestimated if the station is located on the edge of the storm, or overestimated if the station is located near the center of the storm. A relatively large number of precipitation stations is needed for a highly complex watershed. For highly variable flows, precipitation has to be measured more frequently. If the flow forecasts are evaluated by an average criterion, e.g. volume error, a lesser number of precipitation stations can be sufficient; if a more precise characteristic of the hydrograph is needed (e.g. the flow peak), a large number of precipitation stations may be needed.

2.1.4 Reference Data Error

High quality reference data (for example, flow measurements) are mainly used for model calibration and evaluation. In addition, they can be used as a feedback for real-time flow forecasting models. Equipment and human factors are not the only causes of flow measurement errors, since flow rating curves may not represent all of the flow conditions, especially during high flood events.
Chapter 2. Analysis of Errors in Flow Forecasts

2.2 Error Analysis Methods

Several forecast error analysis techniques are presented in this section. Each provides a distinctive view of the differences between measured and forecast hydrographs. Results of these analyses are important in evaluating practical impacts of flow forecast errors, improving model calibration, and designing real-time flow forecasting models.

2.2.1 Graphical Presentations

Many of the characteristics of the flow forecast errors can be easily detected through graphical presentations. Following are some of these plots:

1. Observed and forecast hydrographs (Fig. 2.2).

2. Error signal (Fig. 2.2), which helps in examining the distribution of and spotting extreme values of forecast residuals.

3. Accumulated errors (Fig. 2.2), which is important for evaluating long-term effects of forecast residuals.

4. Observed flow vs. forecast (Fig. 2.3), which is especially important in detecting errors in flow rating curves. Linear plots help in detecting linear relationships between observed and forecast flows. For example, Fig. 2.3a shows that, in general, very small and very large flows are overestimated, while medium range ones are underestimated. Logarithmic plots (Fig. 2.3b) may reveal the presence of nonlinear relationships.

5. Flow forecast residuals versus previous residuals, which is useful for detecting and estimating correlations between consecutive residuals. It can be clearly seen from Fig. 2.4 that a residual is highly correlated with its value one day before. In other
Fig. 2.2 Examples of plots of observed flows, forecast flows, residuals, and accumulated errors.
Chapter 2. Analysis of Errors in Flow Forecasts

Fig. 2.3a Example of a linear plot of observed flow vs. forecast.

Fig. 2.3b Example of a logarithmic plot of observed flow vs. forecast.
words, the residuals can be expressed as follows\(^1\):

\[
r(t) = \alpha r(t - 1) + w(t - 1)
\]

(2.1)

where \(r\) is the forecast residual; \(\alpha\) is a correlation coefficient; and \(w\) is a random noise component.

The autocorrelation and the partial autocorrelation functions, to be discussed later, are more comprehensive methods for evaluating these lagged correlations.

### 2.2.2 Least Squares Criteria

The overall performance of a flow forecasting model can be evaluated against a reference model based on the sum of the squares of residuals. Several criteria can be formulated

\(^1\)This expression describes an autoregressive process of order one, designated as \(AR(1)\).
Chapter 2. Analysis of Errors in Flow Forecasts

depending on the type of the reference model, and the following are some of the most important criteria:

The Coefficient of Efficiency $E$

The coefficient of efficiency, $E$, can be expressed as follows (Nash and Sutcliffe, 1970):

$$E = \frac{S_o - S}{S_o} \quad (2.2)$$

where

$$S = \sum_{i=1}^{N} (Q_o(t) - Q_f(t))^2;$$

$$S_o = \sum_{i=1}^{N} (Q_o(t) - \overline{Q}_o)^2$$

where $Q_o$ is the observed flow;

$\overline{Q}_o$ is the mean of the observed flows for the given record; and

$Q_f$ is the forecast flow.

The quantity $S$ is a direct measurement of the discrepancy between the observed hydrograph and the forecast hydrograph. A high value of $S$ indicates a large discrepancy and therefore a poor forecast, while a small value of $S$ indicates the opposite. $S_o$ is the value of $S$ if the forecast hydrograph is a horizontal line equal to the mean observed flow. The coefficient of efficiency, $E$, is the difference between $S_o$ and $S$ normalized to $S_o$. Therefore, $E$ is a non-dimensional measure of the quality of the flow forecasts. A value of $E$ close to unity indicates a good flow forecast, while a bad flow forecast is characterized by an $E$ value much less than unity.

Choosing $S_o$ as a normalizing factor is based on the well known coefficient of determination, $D$, which will be presented in the next section. Using the same normalizing
factor in both $E$ and $D$ makes it possible to compare between the two coefficients, which will be proven, in a later section, to be useful in estimating systematic errors in the flow forecasts.

The Coefficient of Determination $D$

The coefficient of determination, $D$, can be expressed as follows:

$$D = \frac{S_o - S_r}{S_o}$$  \hspace{1cm} (2.3)

where

$$S_r = \sum_{i=1}^{N} (Q_o(i) - Q_r(i))^2$$

where $Q_r$ is the flow calculated from linear regression of the observed flow against the forecast, so that the criterion $D$ measures the performance of the flow forecasts after removing residuals linearly related to the forecast flow (usually called linear errors.) This leads to the conclusion that the difference between the values of $E$ and $D$ is equal to the normalized sum of squares of linear errors, which leads to the development of a very useful least squares criterion, $PLE$ (percentage of linear errors) which is presented in the next section.

Percentage of Linear Errors $PLE$

Percentage of linear errors, $PLE$, is defined as follows:

$$PLE = (\frac{D - E}{1 - E})100$$  \hspace{1cm} (2.4)

Therefore, $PLE$ is the ratio of the percentage of the sum of squares of linear errors to the sum of squares of all residuals. In other words, $PLE$ measures the improvement in a flow forecast brought about by using linear regression. $PLE$ is not limited to this regression model of $Q_o$ vs. $Q_f$, but can also be used for testing other regression models of observed
flows against other outputs of a flow forecasting model\(^2\). \textit{PLE} can therefore be used to test the validity of real-time flow forecasting models based on these regression models. This will be demonstrated for the flow updating model presented in Chap. 5.

The Coefficient of Persistence \(C_P\)

The coefficient of persistence, \(C_P\), can be expressed as follows (Kitanidis & Bras, 1980c):

\[
C_P = \frac{S_P - S}{S_P}
\]

\[(2.5)\]

where

\[
S_P = \sum_{i=1}^{N}(Q_o(i) - Q_o(i - \Delta t))^2
\]

where \(\Delta t\) is the time interval of flow measurements.

\(C_P\) measures the performance of the flow forecasting model relative to the performance of the naive persistence model, in which the flow is estimated equal to the most recent measured flow. A \(C_P\) value over zero means that the tested model performs better than the persistence model, while a value less than zero indicates the opposite.

Consecutive short-term measured flows are generally highly correlated leading to a good performance of the persistence model for short-term flow forecasts. Therefore, \(C_P\) is useful in evaluating short-term flow forecasts. On the other hand, sparsely measured flows are weakly correlated causing the persistence model to be quite inferior for long-term flow forecasts. For this reason, \(C_P\) is impractical for evaluating long-term flow forecasts.

\(^2\)An example is linear regression of \(Q_o\) vs. \(Q_s\), \(Q_r\) and \(Q_g\) (the snowmelt, the rain and the groundwater flow components, respectively), where

\[
Q_f = Q_s + Q_r + Q_g
\]
The Coefficient of Extrapolation $C_e$

The coefficient of extrapolation, $C_e$, is defined as follows (Kitanidis & Bras, 1980c):

$$C_e = \frac{S_e - S}{S_e}$$

(2.6)

where

$$S_e = \sum_{i=1}^{N}(Q_o(i) - Q_e(i))^2$$

where $Q_e(i)$ is the flow calculated from the extrapolation of the most two recent observed flows. A positive $C_e$ value indicates that the tested model performs better than the extrapolation model; while a negative value indicates the opposite.

Similar to the discussion regarding the coefficient of persistence $C_p$, high correlations among consecutive short-term measured flows make $C_e$ useful for evaluating short-term flow forecasts, while weak correlations among consecutive long-term measured flows make $C_e$ inefficient in evaluating long-term flow forecasts.

Selection of the least Squares Criteria

The quantity $S$ defined in equation 2.2 is a direct measure of the discrepancy between the observed hydrograph and the forecast hydrograph. However, the coefficient of efficiency $E$ is a more suitable quantity for evaluating flow forecasts since it has a small non-dimensional value, and high values of $E$ are associated with good flow forecasts.

Choosing $S_o$ as a normalizing factor for $E$, as compared to $S_p$ and $S_e$ for the coefficient of persistence and the coefficient of extrapolation $C_p$ and $C_e$, respectively, has two main advantages:

1. $S_o$ is a fixed value regardless of the forecast lead, and

2. $E$ can be compared with the coefficient of determination, $D$ to evaluate systematic errors in the flow forecasts.
The coefficient of persistence $C_p$ and the coefficient of extrapolation $C_e$ measure the performances of the persistence model and the extrapolation model, respectively. Since both models are quite powerful for short-term flow forecasts, $C_p$ and $C_e$ are useful in evaluating short-term flow forecasts. However, both criteria are inefficient for the evaluation of long-term flow forecasts, since both the persistence model and the extrapolation model perform very badly for long-term flow forecasts and therefore cannot be used as bases for comparison.

For applications with varying forecast leads, the use of $C_p$ and $C_e$ can be quite confusing, since the normalizing factors $S_p$ and $S_e$, respectively, are different for different forecast leads leading to changes in the values of $C_p$ and $C_e$ even if the flow forecasts have not changed.

It is interesting to note the $C_p$ is proportional to the difference between the coefficient of efficiency calculated for the persistence model and the coefficient of efficiency calculated for the flow forecasting model to be evaluated. This can be proven as follows:

- using Eq. 2.2, the coefficient of efficiency for the persistence model is:

$$E_p = \frac{S_o - S_p}{S_o}$$

(2.7)

- from Eqs. 2.7, 2.2 and 2.5 it can be seen that:

$$C_p = (E - E_p)\frac{S_o}{S_p}$$

(2.8)

The above indicates that using $C_p$ in evaluating the flow forecasts is equivalent to comparing the values of $E$ and $E_p$.

It can be proven, similar to above discussion, that for the coefficient of extrapolation:

$$C_e = (E - E_e)\frac{S_o}{S_e}$$

(2.9)
Chapter 2. Analysis of Errors in Flow Forecasts

where,

\[ E_e = \frac{S_o - S_e}{S_o} \]

The approach of replacing the coefficients of persistence and extrapolation with the quantities, \( E - E_p \) and \( E - E_e \), respectively, has two advantages:

1. It uses one criterion, i.e. the coefficient of efficiency \( E \), in evaluating the flow forecasts.

2. It makes it possible to evaluate flow forecasts against naive models other than the persistence and the extrapolation models.

Based on the discussion in this section, it can be concluded that the coefficient of efficiency \( E \) is most suitable least squares criterion for evaluating flow forecasts. Therefore, \( E \) is chosen to be the main criterion in this research.

Detection and Correction of Timing Errors in the Flow Forecasts

Observed flows can be regressed against lagged calculated flows to detect and estimate timing errors. Values of coefficient of the determination for several lags are compared and the timing error is estimated as the one corresponding to the least value of the coefficient of determination. Lagged regression can also be applied between observed flows and calculated flow components.

Based on this approach a simple updating model can be used to detect, estimate and correct for timing errors in the flow forecasts. The algorithm of this model is presented as follows:

1. Lagged Regression

Multiple linear regression is applied for the past 10 observed flows, \( Q_o \), vs. the lagged snowmelt flow components, \( Q_s \), and the groundwater flow components, \( Q_g \).
calculated by the UBC watershed model. Lags of 3, 2, 1, 0, -1, -2 and -3 days are used. The timing error, \( \lambda \), is estimated equal to the lag that produces the highest coefficient of determination.

2. Prediction

For the next 5 days the flow is estimated equal to a weighted lagged sum of the snowmelt and the groundwater flow components:

\[
Q_u(t) = \alpha_s Q_s(t - \lambda) + \alpha_g Q_g(t - \lambda)
\]  (2.10)

where \( \lambda \) is the estimated timing error calculated from the previous step; and \( \alpha_s \) and \( \alpha_g \) are the coefficients of regression that yielded the highest coefficient of determination.

3. Updating

After obtaining the flow measurements for the next five days, step (1) is repeated to update the estimates of the timing error and the coefficients \( \alpha_s \) and \( \alpha_g \).

The timing error updating model was applied to the flow forecasts generated by the UBC watershed model\(^3\) for the Illecillewaet basin in British Columbia\(^4\) for the period April 15 to July 15, 1970. The measured flows, the non-updated flow forecasts, the updated flow forecasts, and their corresponding residuals are shown in Fig. 2.5. Applying the updating model resulted in a general improvement in the flow forecasts as the coefficient of efficiency, \( E \), increased from 0.74 to 0.82. The improvement in the flow forecasts is limited due to the high instability of the magnitude and the phase of the timing errors.

Although the timing error updating model did not substantially improve the flow forecasts for this particular event, it does provide a simple and logical approach for

\(^3\)The UBC model is described in Chap. 5.

\(^4\)Detailed information of the Illecillewaet basin is presented in Appendix A.
Fig. 2.5 Hydrographs of the observed, the UBC and the timing error-updated flows and their residuals for a forecast lead of five days.
detecting and correcting timing errors. Moreover, this model is a good calibration tool for estimating timing errors in historical data.

2.2.3 The Autocorrelation and the Partial Autocorrelation Functions

The Autocorrelation Function

The autocorrelation function $\rho(\tau)$ measures the correlation between a stochastic variable $x(t)$ and its value $\tau$ time steps ahead $x(t + \tau)$. $\rho(\tau)$ is defined as follows (Priestley, 1987):

$$
\rho(\tau) = \frac{\text{COV}\{x(t), x(t + \tau)\}}{\left[\text{VAR}\{x(t)\}\text{VAR}\{x(t + \tau)\}\right]^{\frac{1}{2}}}
$$

where, $\text{COV}\{x(t), x(t + \tau)\}$ is the covariance of $x(t)$ and $x(t + \tau)$; and $\text{VAR}\{x(t)\}$ is the variance of $x(t)$.

Since covariances and variances are unknown quantities, the autocorrelation function can be estimated from a given sample as follows (IMSL, 1987):

$$
\hat{\rho}(\tau) = \frac{\sum_{t=1}^{N-\tau}[x(t) - \bar{x}][x(t + \tau) - \bar{x}]}{\sum_{t=1}^{N}[x(t) - \bar{x}]^2}
$$

where $\bar{x}$ is the mean; $N$ is the record length; and the hat * indicates an estimate. $\hat{\rho}(\tau)$ is usually called the sample autocorrelation function to differentiate from the theoretical autocorrelation function $\rho(\tau)$) described by Eq. 2.11.

Eq. 2.12 can be used to estimate the correlation between flow forecast residuals at different lags as follows:

$$
\hat{\rho}(\tau) = \frac{\sum_{t=1}^{N-\tau}[r(t) - \bar{r}][r(t + \tau) - \bar{r}]}{\sum_{t=1}^{N}[r(t) - \bar{r}]^2}
$$

The plot of the autocorrelation function vs. the lag is useful in identifying the lags at which the residuals are highly correlated. For example, if the residuals follow an autoregressive process of order one ($AR(1)$), as described below:

$$
r(t) = \alpha r(t - 1) + w(t - 1)
$$
the corresponding autocorrelation function is as shown in Fig. 2.6. On the other hand, the autocorrelation function for an autoregressive process of order two, \((AR(2))\), described by the following equation:

\[
r(t) = \alpha r(t-1) + \beta r(t-2) + w(t-1)
\]

(2.15)
is as shown in Fig. 2.7. Using the residuals of Fig. 2.2 as an example, the sample autocorrelation function is calculated and plotted in Fig. 2.8. The similarity between Fig. 2.8 and Fig. 2.6 indicates that these residuals follow in general an autoregressive process of order one. The bump in Fig. 2.8 indicates the presence of relatively small correlations between residuals with lags between 10 to 16 days. This is more evident in the partial autocorrelation function to be discussed below.

It should be emphasized that the autocorrelation function cannot be used to compare the performances of two different flow forecasting models, since it only evaluates correlations among residuals without any consideration of the quantities of these residuals. This can be seen from Eq. 2.13, which shows that the autocorrelation function is non-dimensional and is not related to the absolute values of the residuals.

**The Partial Autocorrelation Function**

While the autocorrelation function is equal to the correlation between two lagged values of a variable, the partial autocorrelation function is defined as the correlation between these two lagged values less the portion of correlation caused by the influence of correlations among previous values. For example, in Fig. 2.6 the correlations for lags longer than one day are caused by the influence of the correlation for a lag of one day. Removing this influence results in zero correlations for lags longer than one day, and therefore the partial autocorrelation function for an autoregressive process of order one \(AR(1)\) is equal to a single value at a lag of one day and zero for longer lags. Therefore, the partial
Fig. 2.6 Theoretical autocorrelation function of AR(1).

Fig. 2.7 Theoretical autocorrelation function of AR(2).
Fig. 2.8 Example of an autocorrelation function of forecast residuals.

Fig. 2.9 Example of a partial autocorrelation function of forecast residuals.
autocorrelation function is a direct method for identifying the order of an autoregressive process. The partial autocorrelation function can be estimated equal to the diagonal elements of matrix $\tilde{\xi}$ defined as follows (Durbin, 1960):

For $k = 1, \ldots, K$, where $K$ is the maximum possible order of an autoregressive process

$$
\xi(k, k) = \begin{cases} 
\hat{\rho}(1) & k = 1 \\
\frac{\hat{\rho}(k) - \sum_{j=1}^{k-1} \xi(k-1, j)\hat{\rho}(k-j)}{1 - \sum_{j=1}^{k-1} \xi(k-1, j)\hat{\rho}(j)} & k = 2, \ldots, K 
\end{cases}
$$

$$
\xi(k, j) = \begin{cases} 
\xi(k-1, j) - \xi(k, k)\xi(k-1, k-j) & j = 1, 2, \ldots, k-1 \\
\xi(k, k) & j = k.
\end{cases}
$$

The order of an autoregressive process is estimated equal to the argument $m$ of $\xi(m, m)$ beyond which the partial autocorrelation vanishes. Using the residuals of Fig. 2.2 as an example, the partial autocorrelation function is calculated and plotted in Fig. 2.9. It is clear from Fig. 2.9 that the values of the partial autocorrelation function are insignificant beyond a lag of one, hence these residuals follow an autoregressive process of order one.

### 2.3 Summary and Conclusion

Misrepresentative models, miscalibrated parameters, erroneous input and reference data are the four main causes of errors in flow forecasting. Much of the characteristics of these errors can be revealed through various plots of observed and forecast hydrographs, flow forecast residuals, accumulated errors, observed vs. forecast flows, and residuals vs. their lagged values.

Performances of flow forecasting models can also be evaluated by several least squares criteria. The coefficient of efficiency $E$ measures the overall performance of the model; while the coefficient of determination $D$ evaluates the model performance after removing linear errors. Therefore, the difference between these two criteria or more specifically
the newly introduced criterion the $PLE$ is a direct measure of "potentially removable" linear errors. The coefficient of persistence $C_p$ and the coefficient of extrapolation $C_e$ compare the model performance to those of naive models. Based on the discussion in section 2.2.2, the coefficient of efficiency is selected as the primary least squares criterion in this research.

Based on lagged linear regression, a simple updating model was developed to detect, estimate and correct for timing errors in flow forecasts. Applying this updating model to flow forecasts generated by the UBC watershed model resulted in limited improvement in the flow forecasts. The inability of the updating model to significantly improve the flow forecasts is due to the high variability of the timing errors in the flow forecasts. However, this updating method is a useful tool for detecting and estimating timing errors in historical data.

The degree and persistence of correlation among residuals can be estimated using the autocorrelation function. The partial autocorrelation function is a more direct method for estimating the order of an autoregressive process. In spite of being excellent tools for analyzing error signals, these function do not take into consideration the size and distribution of residuals and therefore cannot be used for comparing the performances of different models.
Chapter 3

Approaches to Real-Time Flow Forecasting: A Literature Review

As has been stated earlier, the main objective of this research is to design methods for updating the flow forecasts of watershed models using real-time (on-line) data. These methods belong to the group of real-time flow forecasting models, which are defined as models that use real-time reference data (for example, flow or snowpack measurements) as well as input data in calculating flow forecasts. The updating procedures in real-time flow forecasting models follow two main approaches:

1. Updating flow forecasts generated from a time-invariant (off-line) model; and
2. Adjusting variables (parameters, water storages, etc.) of a time-variant (on-line) flow forecasting model to achieve a better correspondence between measured and forecast hydrographs.

The real-time models adopting the first approach are not flow forecasting models per se, since they are basically procedures designed to correct the flow forecasts of given flow forecasting models. Their applications do not necessitate altering the structures of the corresponding flow forecasting models. These real-time models can be classified into two types:

- Error prediction models (section 3.1), and
- Flow forecast correction models (section 3.2).
Chapter 3. Approaches to Real-Time Flow Forecasting

The real-time models of the second approach are dynamic models specifically designed for the purpose of updating their variables from real-time data. These models can be classified according to the nature of their structures as follows:

- Transfer function models (section 3.3); and
- Conceptual models (section 3.4).

In the following sections the above four types of real-time models are discussed with more emphasis on the last two types (i.e., the transfer function and the conceptual models).

3.1 Error Prediction Models

In many flow forecasting applications, residuals tend to be highly correlated with their past values. Using techniques established in the time series analysis literature (Box and Jenkins, 1970) magnitudes and extensions of these correlations can be evaluated. Corresponding algorithms can then be designed to simulate relationships between consecutive residuals and these can be used in estimating future residuals. Examples of these models can be seen in (Szöllösi-Nagy, 1984; Moore, 1982; Jones, 1980; and Marivoet, 1980). A very common example is the autoregressive (AR) model (Moore, 1986):

$$r(t) = -\phi_1 r(t-1) - \phi_2 r(t-2) - ... - \phi_z r(t-z) + w(t-1)$$ (3.1)

where $r(t) = Q_f(t) - Q_o(t)$ is the forecast residual at time $t$, and $Q_f(t)$ and $Q_o(t)$ are the forecast and observed flows, respectively;

$\phi_1, \phi_2, ... \phi_z$ are model parameters; and

$w$ is a random noise of zero mean and known variance $\sigma_w^2$. 
At time $t - 1$, the future residual $r(t)$ can be estimated equal to the right hand-side of Eq. 3.1 less $w(t - 1)$, which is best estimated equal to zero\(^1\). Model parameters can be estimated through calibration using historical data. In simple applications these estimates are kept fixed during the forecasting period. However, a more realistic approach is to update these estimates as new data becomes available using one of many recursive estimating schemes available in the literature (e.g., recursive regression, maximum likelihood).

Error prediction models are flexible and easy to apply. They are especially useful in providing feedback of flow measurements to highly complex off-line flow forecasting models. However, their performance is highly dependent on the persistence of forecast residuals. As noted by Moore (1986), this persistence is least in the vicinity of the hydrograph rising limb, while it is very strong on the falling limb. Unfortunately, this situation results in poor performance where it is needed the most (i.e., on the rising limb) and good performance where it matters the least (i.e., on the falling limb). As will be seen, this problem is also observed by other researchers working on other types of real-time forecasting models (i.e, the transfer function and the conceptual models). Timing errors are another serious obstacle in applying error prediction models. A simple method for detecting and correcting timing errors was presented in section 2.2.2.

3.2 Flow Forecast Correction Models

A comparative analysis between measured and forecast hydrographs (through graphical plots or regression analysis) may reveal certain relationships between the measured and the forecast flows. This may be in the form of a bias or even a nonlinear relationship. Based on the results of these analyses, a model can be designed to update the flow

\(^1\)Since its mean is equal to zero.
forecasts using real-time flow data. A simple example is the flow updating model that will be presented in Chap. 4, in which the updated flow is calculated as a linear function of the flow forecast calculated from an off-line flow forecasting model.

The forecast flow can also be updated as a function of outputs from an off-line flow forecasting model. An example is the flow updating model to be presented in Chap. 5, in which the flow is calculated as a weighted sum of the calculated flow components (snowmelt, rainfall and groundwater). The weighting factors are recursively updated from on-line flow measurements using the Kalman filter, which will be discussed in Chap. 4.

3.3 Transfer Function Models

Frequently referred to as black box or input-output models, transfer function models became popular after the advent of the text “Time Series Analysis” by Box and Jenkins (1970). In these models the flow is estimated as a weighted sum of the past observed flows and the input data (precipitation, temperature, etc.). An example is the following model (Moore, 1986):

\[ Q_p(t + 1) = \alpha_0 Q_o(t) + \beta_0 p_r(t) + \beta_1 p_r(t - 1) + \beta_2 p_r(t - 2) \] (3.2)

where \( Q_p \) is the predicted flow;

\( Q_o \) is the observed flow;

\( p_r \) is the effective rainfall; and

\( \alpha_0, \beta_0, \beta_1 \) and \( \beta_2 \) are weighting factors.

The above model has been proven (Moore, 1986) to be equivalent to the following unit hydrograph model:

\[ Q_p(t) = v_0 p_r(t) + v_1 p_r(t - 1) + v_2 p_r(t - 2) + ... \] (3.3)
where \( v_0, v_1, ..., v_n \) are the ordinates of the unit hydrograph.

The principle of continuity requires that:

\[
\beta_0 = 1 - (\alpha_0 + \beta_1 + \beta_2)
\]  \hspace{1cm} (3.4)

Moore (1986) points out that the accuracy of estimating the effective rainfalls is very critical for the success of the transfer function model. He also states that this problem is equivalent to choosing an appropriate loss model in the unit hydrograph model.

Most transfer function models are formulated in a state-space format on which the Kalman filter can be applied. Following are presentations and discussion of two transfer function models selected from the literature:

1. (Todini, 1978b):

\[
Q_p(t + 1) = \alpha_0 Q_p(t) + \alpha_1 Q_p(t - 1) + \beta_0 p_d(t) + \\
\beta_1 p_d(t - 1) + \beta_2 p_d(t - 2) + \gamma_0 p_w(t) + \\
\gamma_1 p_w(t - 1) + \gamma_2 p_w(t - 2)
\]  \hspace{1cm} (3.5)

where

\[
\begin{align*}
p_d(t) &= pp(t) \quad \text{For dry soil conditions;} \\
p_w(t) &= 0
\end{align*}
\]

\[
\begin{align*}
p_d(t) &= 0 \\
p_w(t) &= pp(t) \quad \text{For wet soil conditions}
\end{align*}
\]

where \( pp \) is the total precipitation.

Soil moisture condition is determined by the antecedent precipitation index (API) calculated as a function of time:

\[
API(t) = \theta API(t - 1) + pp(t)
\]  \hspace{1cm} (3.6)
Chapter 3. Approaches to Real-Time Flow Forecasting

where \( \theta \) is a constant. If \( API \) is less than a threshold value of \( T \), the soil is considered dry; otherwise the soil is considered wet. The model takes into consideration the effect of soil conditions on the rainfall-runoff relationship by assigning small values for the parameters \( \beta_0, \beta_1 \), and \( \beta_2 \) and larger values for the parameters \( \gamma_0, \gamma_1 \), and \( \gamma_2 \). This causes the precipitation at dry conditions, \( p_d(t) \), to be passed to the flow more slowly than the precipitation at wet conditions, \( p_w(t) \).

The flow is estimated through applying the mutually interactive state-parameter estimation (MISP) technique (Todini, 1976) on a state-space format of Eq. 3.5. The MISP is basically an adaptive Kalman filter method.

The model shows a satisfactory performance for a forecast lead of two hours. Close examination of the measured and the forecast hydrographs shows that although the flow forecast of the falling limb is good up to a forecast lead of six hours, model performance on the rising limb is quite poor beyond a forecast lead of four hours. This comes as no surprise, since Eq. 3.5 shows that the forecast flow is dependent on the correlation of the flow with the two most recent flows. This correlation is strong and consistent for the falling limb; while it is the weakest on the rising limb. This problem is shared by all transfer function models.

2. (Burn, 1985):

\[
Q_p(t + 1) = \alpha_0 Q_\sigma(t) + \alpha_1 Q_\sigma(t - 1) + \beta_0 pp(t) + \beta_1 pp(t - 1) + \\
\beta_2 pp(t - 2) + \beta_3 pp(t - 3) + \beta_4 pp(t - 4) + \gamma_0 T(t) + \\
\gamma_1 T(t - 1) + \gamma_2 T(t - 2) \tag{3.7}
\]

\[\text{Adaptive Kalman filtering will be discussed in section 4.4.}\]
where $T(t)$ is the temperature index for snowpack melting represented by:

$$m(t) = d(t)T(t)SCA(t)$$ \hspace{1cm} (3.8)

where $m(t)$ is the snowmelt at time $t$; $d(t)$ is a degree-day factor relating the temperature index to the depth of water column produced on that day; and $SCA(t)$ is the snow covered area as percentage of the entire basin area.

The MISP technique, mentioned in the previous example, has been applied on a state-space format of Eq. 3.7. In a later work (Burn, 1988), the above model has been modified by including time index and precipitation forecasts. In both versions the model performance was only acceptable for short forecast leads (one day or less). Longer forecast leads (two and three days) show a dramatic deterioration in flow forecasts. This is typical of transfer function models.

Both of the transfer function models presented above share the following properties that can be generalized for this class of real-time flow forecasting models:

1. Their dependency on data feedback give them the ability to perform quite well for very short forecast leads; while their lack of physically-based structures eliminate their use for long- or even medium- forecast leads.

2. While the dependency on the correlation among consecutive flows is the main factor for their exceptional performance on the hydrograph falling limb, it is the same reason for inferior forecasts on the rising limb.

3. The simplicity of these models make them easy and flexible to apply. In addition less time and effort are needed in calibration and operation.
3.4 Conceptual Models

The term conceptual describes those models designed deterministically to simulate physical processes of the watershed system. In these models the soil is visualized as a network of water storages. Input of rainfall and/or snowmelt, less losses, are split among some of these storages according to a priority system. Excesses of water from high priority storages are passed down to lower priority storages, and so on. The outflow from each of these storages is assumed to be a function of the storage itself \( Q = F(S) \). These outflows may then be passed directly to the basin outlet (fast flow) or may be routed through a series of storages before reaching the basin outlet (e.g., ground water). Fig. 3.1 shows the structure of a basic conceptual model.

The nonlinear nature of conceptual models and the presence of large number of variables poses a difficult problem in adapting these models to the real-time mode. This problem can be solved by linearizing model equations (e.g., through Taylor expansion) and selecting only sensitive variables for updating. Two examples of real-time conceptual models are discussed below:

1. (Kitanidis and Bras, 1980 (b & c))

The National Weather Service Catchment Model (NWSC) has been linearized using the method of statistical linearization (Gelb, 1986) for threshold-type equations and Taylor expansion for the rest. The Kalman filter is applied on a state-space format of the linearized model to update eight variables (states): five soil moisture storage elements, impervious area, channel storage, and the coefficient of the linear reservoir routing scheme.

From reviewing the results of several runs of this model, the following has been observed and concluded:
Fig. 3.1 Flow chart of a basic conceptual model.
• The model is essentially sensitive to three variables only: the lower zone secondary soil storage (fast base flow), the channel storage and the coefficient of linear reservoir routing scheme. The model performance was greatly improved when the state noise variances associated with these variables were increased.

• Supplementing the model with a scheme for detecting isolated errors in variables (developed by Kitanidis and Bras (1980a)) resulted in slightly inferior performance for short forecast leads (up to 18 hours) and marginal improvement in performance for longer forecast leads (up to 36 hours).

• Comparing the performance of this conceptual model with those of naive models (persistence and extrapolation) and a black box model shows that although there are no essential differences in performance for very short forecast leads (six hours), the conceptual model is much superior to the other models, especially the extrapolation and black box models, for longer forecast leads.

• Model performance deteriorates rapidly for longer forecasts leads. The coefficient of efficiency $E$ (defined in Chap. 2) lost about 30% of its value as the forecast lead is increased from 6 hours to 36 hours. This creates some doubts about the validity of this model for use in long-term flow forecasting.

• The authors raise the most serious problem in real-time forecasting: i.e., predicting the beginning of the rising limb and the magnitude and timing of the peak. The rising limb is driven by the storage in the upper zone (surface runoff) which is unobservable until it starts to occur, by which time there is little time left for the model to update its states before the flow reaches its peak.

• Timing errors may lead to erroneous updating of the variables.

• It is noted that the period used for testing model performance is the same one
used for selecting the state noise variance. Therefore, the results shown do not represent a “true” real-time test for the model.

2. (Georgakakos and Bras, 1986 (a & b))

Three physically-based models have been integrated into a single state-space model. A precipitation model (Georgakakos, 1984 (a & b)) yields real-time precipitation forecasts from inputs of temperature, pressure and dew point temperature. Then, precipitation input and flow measurements are processed through a soil moisture accounting model (Kitanidis and Bras, 1980 (a & b)) to estimate the flow at the basin outlet. Lastly, the flow is routed through a nonlinear channel routing model (Georgakakos, 1982) to calculate downstream flows. So, this model is basically a more developed version of the model discussed in the previous section (Kitanidis and Bras, 1980 (b & c)).

Although precipitation forecasts generated by this model are found to be quite poor, they had little impact on the quality of flow forecasts, indicating that errors in precipitation may have been offset by adjusting soil moisture storages.

In spite of the model’s physical background and comprehensiveness it was outperformed by naive models (persistence and extrapolation) for forecast leads of 6 and 12 hours. The conceptual model did better than the naive models for longer forecast leads (18, 24 and 30 hours), however its performance deteriorated very rapidly as the forecast lead was increased. So, in general, this model is not expected to be reliable for longer forecast leads.
3.5 Summary and Conclusion

Real-time flow forecasting models are classified into four types.

Error prediction and flow forecast correction models are based on adjusting flow forecasts generated from off-line flow forecasting models. They are simple and flexible approaches for integrating real-time data into complex and rigid flow models.

Transfer function and conceptual models are designed, mostly in a state-space format, to exploit real-time data in updating some of the model variables. Transfer function models are based on the input-output relationship without consideration to the driving forces behind such relationship; while the conceptual models are designed to simulate the physical processes in the watershed system.

Transfer function models are suitable for very short forecast leads. However, their dependency on feedback makes them unreliable for longer forecast leads. In such situations conceptual models are considered to be better alternatives. Unfortunately, performances of conceptual models deteriorates rapidly for relatively long forecast leads (beyond two days\(^3\)).

All types of real-time flow forecasting models are observed to be superior on the falling limb of the hydrograph. Unfortunately, forecasting on the rising limb is generally very poor. For the transfer function models this is attributed to the correlation for consecutive flows, which is strong on the falling limb and weak on the rising limb. For conceptual models, the falling limb is controlled by the water storage of the base flow which changes very slowly and therefore is easier to update; the rising limb is driven by the water storages of the surface flow and the interflow which are usually too variable so that the updating routine cannot keep track of these changes.

In addition to the errors discussed, the real-time models are also susceptible to timing

\(^3\)As observed from the applications of the two models discussed in this chapter.
errors. All the above models lack any scheme to detect such errors. A simple method to solve this problem was presented in section 2.2.2.
Chapter 4

The Kalman Filter

As has been noticed from Chap. 3, most of the updating procedures used in the real-time flow forecasting models are based on the Kalman filter technique. The popularity of this technique, not only in hydrological forecasting but in many other engineering applications, is due to its ability to provide a feedback from on-line data to update model’s estimates in an easily applicable recursive approach. The main features and an example of the application of the Kalman filter to updating flow forecasts are presented and discussed in this Chapter.

4.1 Example of a Kalman Filter Application

An attempt to read the original work of Kalman (1960, 1961) or many of the publications concerning the theoretical background of the Kalman filter may be discouraging for many hydrologists in pursuing a full understanding of this approach and its potential applications in hydrology. This situation stems from the unfamiliar terminology and the relatively advanced statistical concepts used in the literature. To avoid this problem, the Kalman filter is presented through a simple example of its application in updating flow forecasts. The example sheds light on some of the practical features of the Kalman filter, leading the way to discuss important aspects and problems in applying this technique.

In this example, daily flow at a basin outlet is being forecast by a watershed model, and a Kalman filter-based procedure is designed to update these flow forecasts for different forecast leads.
4.1.1 Conditions

The Kalman filter was developed based on the assumption that the variable to be updated, referred to as the state, is normally distributed and is equal to a linear function of its past values and a normally distributed random error. In addition, the measurement is assumed to be a linear function of the state and a normally distributed random error. Based on these assumptions, the following conditions are set for the application of the flow updating model.

1. The observed flow is a linear function of the flow calculated by the watershed model:

\[ Q_o(t) = \alpha(t)Q_e(t) + \nu(t) \]  

where \( Q_o \) is the observed flow; \( Q_e \) is the flow estimated by the watershed model; \( \alpha \) is a correction factor, referred to as the state; and \( \nu \) is a random noise of zero mean and variance \( \sigma^2_\nu \) (referred to as the measurement variance).

Eq. 4.1 is referred to as the Measurement equation, since it describes the relationship between the measurement and the state. The flow measurements are available on intervals of \( \Delta t \).

2. The state \( \alpha \) is related to its past value as follows:

\[ \alpha(t_k) = \alpha(t_{k-1}) + w(t_{k-1}) \]  

where \( t_k - t_{k-1} = \Delta t \); and \( w \) is a random noise of zero mean and variance equal to \( \sigma^2_w \). The variance \( \sigma^2_w \) is referred to as the state noise variance. Eq. 4.2 is referred to as the State equation, since it describes the structure of the state.

3. The noises \( \nu \) and \( w \) are uncorrelated. This is a reasonable assumption, since the method of measuring the flow, and therefore the accuracy of the flow measurement is independent of the method used in estimating the flow forecasts.
Chapter 4. The Kalman Filter

4.1.2 The State-Space Model

The combined set of the state and the measurement equations (Eqs. 4.2 and 4.1, respectively) are usually referred to as the state-space model, which can be written in the following conventional matrix format of the Kalman filter literature:

\[
\begin{align*}
    x_k &= x_{k-1} + w_{k-1} \quad \text{State Equation} \\
    z_k &= H_k x_k + v_k \quad \text{Measurement Equation}
\end{align*}
\] (4.3)

where, \( x_k = [\alpha(t_k)] \); \( w_k = [\omega(t_k)] \); \( z_k = [Q_o(t_k)] \); \( H_k = [Q_e(t_k)] \); and \( v_k = [v(t_k)] \). The state-space format is written in the matrix format, since in general, the state vector \( x_k \) may be composed of several states, as in the flow updating model presented in Chap. 5, and \( z_k \) can be a vector of several measurements (e.g., rain, flow, snowpack etc.).

The state-space model is a convenient format for the application of the Kalman filter for the following reasons:

- It explicitly presents the two sources from which the state is estimated, i.e. its past values and the measurements.
- Its time-domain format is suitable for the application of recursive approaches like the Kalman filter.
- It explicitly presents the stochastic characteristics of the measurement and the state.

4.1.3 Model Algorithm

The following is the algorithm of the model for updating flow forecasts:

1. Prediction before a measurement is available.

   The state \( x_k \) can be estimated from the state equation (Eq. 4.3) as follows:

   \[
   \hat{x}_k(-) = \hat{x}_{k-1}(+) \quad (4.4)
   \]
where the hat "\(^\wedge\)" indicates that the quantity is an estimate; the negative sign "\(^-\)" means that the estimate is calculated before the measurement is available; and the positive sign "\(^+\)" means that the estimate is calculated after the measurement becomes available. The state noise \(w_{k-1}\) disappeared from Eq. 4.4, since its best estimate is its mean, which is equal to zero.

Another important quantity associated with the state is its variance \(\sigma_k^2\), which is estimated using the following equation:

\[
P_k(-) = P_{k-1}(+) + G_{k-1}
\]

where \(P_k = [\sigma_k^2(t_k)]\) and \(G_k = [\sigma_{w}(t_k)]\). The state variance \(P_k(-)\) is important for estimating the confidence limits of \(\hat{x}_k(-)\).

2. Updating.

Once a new flow measurement becomes available (at time \(t_k\)), the state \(x_k\) can be estimated from the measurement equation (Eq. 4.3) as follows:

\[
H_k \hat{x}_k^o = z_k
\]

leading to:

\[
\hat{x}_k^o = \left[ \frac{Q_{\alpha}(t_k)}{Q_{\varepsilon}(t_k)} \right]
\]

where, the small circle "\(^o\)" indicates that the estimate is obtained from the measurement only. The measurement noise \(v_k\) does not appear in the above equation since its best estimate is its mean, which equals zero.

\(x_k(+)\) can then be estimated as a weighted sum of the estimates from Eqs. 4.4 and 4.6 as follows:

\[
\hat{x}_k(+) = [I - S_k] \hat{x}_k(-) + S_k \hat{x}_k^o
\]
Chapter 4. The Kalman Filter

The Kalman filter is based on deriving the value of the gain $S_k$ that optimizes the estimate $\hat{x}_k$. An optimal estimate is one that is unbiased\(^1\) and yields minimum sum of squares of forecast residuals. According to these conditions the gain $S_k$ can be expressed as follows:

$$S_k = K_k H_k \quad (4.8)$$

where $K_k$ is the Kalman gain defined as follows:

$$K_k = \hat{P}_k(-) H_k^T [H_k \hat{P}_k(-) H_k^T + R_k]^{-1} \quad (4.9)$$

where $R_k = [\sigma_0^2(t_k)]$.

In the literature of the Kalman filter, the Kalman gain $K_k$ is used instead of the gain $S_k$, since Eq. 4.7 is usually expressed in this literature as follows:

$$\hat{x}_k(+) = \hat{x}_k(-) + K_k(z_k - H_k \hat{x}_k(-)) \quad (4.10)$$

where $H_k \hat{x}_k(-)$ is the estimate of the measurement $z_k$ (see the measurement equation, Eq. 4.3,) and $r_k$ is the forecast residual.

Eq. 4.10 has a very important computational significance, since for multiple state applications (for example, the flow updating model presented in Chap. 5) the state estimate from the measurement only $\hat{x}_k^o$ (Eq. 4.6) can not be uniquely determined, and therefore Eq. 4.7 can not be used. However, Eq. 4.7 can be applied to single state applications (for example, the flow updating model presented in this chapter), and is particularly useful in illustrating explicitly that the state is estimated from two independent sources of information, namely: the state equation, and the measurement equation. Moreover, the gain $S_k$ is a non-dimensional matrix; whereas,

\(^{1}\)An unbiased estimate $\hat{x}_k(\pm)$ is one that converges to the actual $x_k$ as more measurements become available.
the Kalman gain $K_k$ is of units related to the given application, for example the units of the Kalman gain for the example presented in this chapter are $\frac{\text{m}}{\text{s}}$.

The estimate of the state variance $\hat{P}_k$ is updated by the new measurement as follows:

$$\hat{P}_k(+) = [I - S_k]\hat{P}_k(-) \quad (4.11)$$

3. Prediction.

The state value for the next time interval, $\Delta t$, can be estimated using Eq. 4.4 as follows:

$$\hat{x}_{k+1}(-) = \hat{x}_k(+) \quad (4.12)$$

The state variance for the next interval, $\Delta t$, can be estimated using Eq. 4.5 as follows:

$$\hat{P}_{k+1}(-) = \hat{P}_k(+) + G_k \quad (4.13)$$

The measurement $z_{k+1}$ can be estimated using the measurement equation as follows:

$$\hat{z}_{k+1} = H_k\hat{x}_{k+1}(-)$$

$$= [Q_u(t_{k+1})]$$

where $Q_u(t_{k+1}) = Q_e(t_{k+1})\alpha(t_{k+1})$ is the updated flow forecast for time $t_{k+1}$.

Variance of $z_{k+1}$ is estimated as follows:

$$\text{VAR}(\hat{z}_{k+1}) = H_{k+1}\hat{P}_{k+1}(-)H_{k+1}^T \quad (4.15)$$

It is interesting to note from Eqs. 4.8 and 4.15 that:

$$H_kK_k = \text{VAR}(\hat{z}_k)[\text{VAR}(\hat{z}_k) + R_k]^{-1} \quad (4.16)$$
The above equation shows that $K_k$ is a function of the ratio of the measurement prediction variance $\text{VAR}(\hat{z}_k)$ to the sum of the measurement prediction variance and the measurement noise variance $R_k$. Applying this equation to the flow updating model discussed shows that the gain $S_k$ is equal to this ratio:

$$S_k = \left[ \frac{\text{VAR}(Q_u(t_k))}{\text{VAR}(Q_u(t_k)) + \sigma^2_u} \right]$$  \hspace{1cm} (4.17)

Since the variance is a measure of accuracy, it can be seen that the contributions of the state and the measurement equations in estimating the state are weighted according to their relative accuracies. A good measurement (small $R_k$) as compared to a poorly structured state equation (large $G_k$ and therefore large $\text{VAR}(\hat{z}_k)$) results in large $S_k$ giving more weight to the measurement in estimating $x_k$ (see Eq. 4.7). An opposite situation results in small $S_k$ and therefore a small contribution of the measurement in estimating $x_k$ (see Fig. 4.1.)

4.2 The Kalman Filter as a Bayesian Approach

The Kalman filter can be looked upon as a Bayesian approach which updates the normal probability density function (expressed by its mean and variance) of a state using relevant measurements. The mean, $\hat{x}_k(-)$, and the variance, $\hat{P}_k(-)$, of the a priori probability density function of the state $x_k$ are updated to become the mean, $\hat{x}_k(\cdot)$, and the variance, $\hat{P}_k(\cdot)$, of the a posterior probability density function using Eqs. 4.7 and 4.11, respectively (see Fig. 4.2.)

As in any-Bayesian approach, the extent of updating an estimate is dependent on the accuracy of the measurement. A good measurement results in significant updating; while

---

3This is only true for the applications of a single state and a single measurement. For multiple state and/or multiple measurement applications, $S_k$ is a matrix linearly related to this ratio.

3Normal probability density function is comprehensively described by its mean and variance.
Fig. 4.1a Effect of a good measurement and a poorly structured state equation on updating.

Fig. 4.1b Effect of a bad measurement and a well-structured state equation on updating.
Chapter 4. The Kalman Filter

"priori" pdf

NEW MEASUREMENT

THE KALMAN FILTER

"posterior" pdf

Fig. 4.2 The Kalman Filter as a Bayesian Approach.

a bad one results in minimal updating. This statement is parallel to the one stated at the end of the previous section.

4.3 The Roles and The Selection of The State Noise Variance and The Measurement Variance

Eq. 4.8 shows that the gain $S_k$ is a function of the relative values of the state variance $\hat{P}_k(-)$ and the measurement variance $R_k$. Since, according to Eq. 4.5, $\hat{P}_k(-)$ is the sum of $\hat{P}_{k-1}(+)$ and the state noise variance $G_{k-1}$, it can be concluded that the gain $S_k$ is a function of the relative values of $\hat{P}_{k-1}(+)$, $R_k$ and $G_{k-1}$. From the recursive Eqs. 4.5 and 4.11, it can be seen that the state variance $\hat{P}_k(+)$ is a function of the initial state variance $\hat{P}_0$, the state noise variance and the measurement variance from time $t = 0$ to $t = t_k$. From the above discussion, it can be finally concluded that the updating process is controlled by the relative values of the initial value of the state variance, the state
noise variance and the measurement variance. In the vicinity where the effects of the initial conditions are insignificant, the updating process is therefore only controlled by the relative values of the state noise variance and the measurement variance. Optimal values of these variances can be estimated from sensitivity analysis of the historical record. This procedure has been applied for the flow updating model presented in Chap. 5.

A simulation study based on the example presented in the previous sections shows that three factors have to be considered in selecting the measurement variance and the state noise variance, namely: the rate of change of the state $x$, the measurement quality and the forecast lead.

In this study, the actual flows are assumed to be the ones represented by the solid line in Fig. 4.3, and the measured flows are simulated according to Eq. 4.1 using a fixed measurement variance $\sigma^2$ equal to $400 \left(\frac{m^3}{s}\right)^2$ (based on the assumption that the flow measurement is within the range of $\pm 20 \frac{m^3}{s}$). These simulated flows and the corresponding errors (measurement errors) are also shown in Fig. 4.3. The state $\alpha(t)$ is selected as follows (see Fig. 4.4):

$$
\alpha(t) = \begin{cases} 
1.5, & 1 \leq t \leq 20, \\
41 \leq t \leq 60, & 81 \leq t \leq 92 \\
3.0, & 21 \leq t \leq 40, & 61 \leq t \leq 80 
\end{cases}
$$

The state values are assumed to be unknown and are to be estimated by the Kalman filter applied on the state-space model described by Eq. 4.3. Fig. 4.4 shows the results of two runs of different values of the state variance for a forecast lead of one day. For a very small state noise variance ($\sigma^2 = 1E-6$), the model was slow in response to the changes in the state value; while for larger state noise variance ($\sigma^2 = 0.01$), model performance was greatly improved due to the increase in the flexibility in tracking the changes in the state. However, running the model for the latter state noise variance of 0.01, but for longer forecast lead of 5 days resulted in severe deterioration in model performance (Fig. 4.4).

This example is similar to the example presented by Ljung and Söderström (1987, P. 275-278).
Fig. 4.3 Actual and measured flows and measurement errors. (Simulated data)
Fig. 4.4 Actual and predicted states for a forecast lead of one day.
This problem was solved by increasing the state noise variance to 0.1 (Fig. 4.5). These results can be explained as follows: since quite few measurements, 19 as compared to a total of 92, were available for updating for a forecast lead of 5 days, the chances of updating the state estimates were greatly reduced, resulting in the poor performance for the run of a state noise variance of 0.01. Increasing the state noise variance to 0.01 results in greater degree of updating for every forecast increment, and therefore improving model performance.

A guide for selecting the state noise variance with respect to the forecast lead was developed as follows:

For a forecast lead of one day, \( \Delta t = 1 \), the state variance for time \( t_{k+1} = t + 1 \) can be estimated using Eq. 4.13 as follows:

\[
\begin{align*}
\hat{P}_{k+1}(-) &= \hat{P}_k(+) + G_k \\
\hat{\sigma}_x^2(t+1)_- &= \hat{\sigma}_x^2(t)_+ + \sigma_w^2(t)
\end{align*}
\] (4.18)

If no measurement is available at time \( t + 1 \), the state variance is not updated, i.e:

\[
\begin{align*}
\hat{P}_{k+1}(+) &= \hat{P}_{k+1}(-) \\
\hat{\sigma}_x^2(t+1)_+ &= \hat{\sigma}_x^2(t+1)_-
\end{align*}
\] (4.19)

The state variance for time \( t + 2 \) is estimated using Eq. 4.13 as follows:

\[
\begin{align*}
\hat{\sigma}_x^2(t+2)_- &= \hat{\sigma}_x^2(t+1)_+ + \sigma_w^2(t+1) \\
&= \hat{\sigma}_x^2(t+1)_- + \sigma_w^2(t+1) \\
&= \hat{\sigma}_x^2(t)_+ + \sigma_w^2(t) + \sigma_w^2(t+1)
\end{align*}
\] (4.20)

If no measurement is available for the next \( n \) days, then:

\[
\begin{align*}
\hat{\sigma}_x^2(t+n)_- &= \hat{\sigma}_x^2(t)_+ + \sum_{i=0}^{n-1} \sigma_w^2(t+i)_{(\Delta t=1)}
\end{align*}
\] (4.21)
Fig. 4.5 Actual and predicted states for a forecast lead of five days.
Chapter 4. The Kalman Filter

For a forecast lead of \( n \) days, \( \Delta t = n \), Eq. 4.13 can be written in the following scalar format:

\[
\hat{x}_n(t + n) = \hat{x}_n(t) + \sigma_w(t)(\Delta t = n)
\]  

(4.22)

Comparing Eq. 4.22 with Eq. 4.21 it can be seen that:

\[
\sigma^2_x(t)(\Delta t = n) = \sum_{i=0}^{n-1} \sigma^2_w(t + i)(\Delta t = 1)
\]  

(4.23)

So, for the example given above, if \( \sigma^2_w(\Delta t = 1) = 0.01 \), then the state noise variance for a forecast lead of five days is selected as follows:

\[
\sigma^2_w(\Delta t = 5) = \sum_{i=0}^{4} \sigma^2_w(t + i)(\Delta t = 1) = 5[\sigma^2_w(\Delta t = 1)] = 5(0.01) = 0.05
\]

Caution should be taken, however, in selecting a very large value for the state noise variance, since a very large state noise variance may cause extreme fluctuations in the state estimates. This can be seen in Fig. 4.6, where running the model for a very large state noise variance of 0.5 results in large fluctuations in the state estimates, especially at the first third of the run period. It is interesting to note, however, that on the second half of the run these fluctuations diminished and the state estimates were very close to the ones produced for a much smaller variance of 0.01. These results can be explained as follows: since the measured flows were simulated based on a fixed measurement variance \( \sigma^2_v \) equal to 400 \( \left( \frac{m^3}{s} \right)^2 \), which is equivalent to a standard deviation \( \sigma_v \) equal to 20 \( \frac{m^3}{s} \), the measurement errors of the small flows (below 50 \( \frac{m^3}{s} \)) were proportionately much larger than those of much larger flows. Since the first third of the forecast period is characterized by small flows, the state estimates based on the measurement only, i.e. \( \hat{x}_k^o \) (Eq.4.6),
Fig. 4.6 Actual and predicted states for state noise variances of 0.01 and 0.5.
were quite poor for the early stage of the forecast period; while they were much better for the following interval of high flows (see Fig. 4.7.) However, since the measurement contribution to the net estimate of the state (i.e., from both the measurement and the previous estimate of the state) is controlled by the gain $S_k$, according to Eq. 4.7, where the larger the value of $S_k$ the more the contribution of the measurement and vice versa, the effects of the early poor measurements on the state estimate for the run of a state noise variance of 0.01 were minimal since the corresponding values of the gain $S_k$ were small (see Fig. 4.8), whereas the effects of these poor measurements on the states of the run of a state noise variance of 0.5 were quite significant since the corresponding values of the gain $S_k$ were quite large (see also Fig. 4.8.) The above discussion shows the importance of checking the quality of measurements if large state noise variances are to be used.

Small state noise variance are preferred for slow time-varying states. For example, if the actual states have a fixed value of 2.25 (see Fig. 4.9), running the model for a small state noise variance value of 0.01 yields good results. Further reduction of the state noise variance to a much smaller value of 1E-6 results in more consistent results beyond the zone of the influence of initial conditions.

From the above results, the following can be concluded about the selection of the ratio of the state noise variance to the measurement variance:

- It should be high enough to track the variability of the state.

- Relatively larger values are preferable for longer forecast leads. Eq. 4.23 can be used as a guide for selecting the state noise variance.

- Very large values should be avoided, unless the measurements are of a very high quality. This is so, since a very large state noise variance causes the updating model to be highly sensitive to errors in the measurements.
Fig. 4.7 Actual states and the states estimated by the measurement only.
Fig. 4.8 The values of the gain $S$ for two state noise variances.
Fig. 4.9 Actual and predicted states for a slow-time varying state.
• Small values are recommended for slow-time varying states.

• The key point in selecting this ratio is to keep a balance between the model’s ability to track changes in the state (model alertness) and minimizing model instability.

From the simulation results it can be argued that the relative values of the measurement and the state noise variances can be treated as design parameters that control the updating process in the Kalman filter. This involves selecting the optimal values that enable the model to be “alert” to true changes in state values, while ignoring changes caused by random errors. Since the states in many flow forecasting applications are unobservable, it becomes necessary to test the model performance according to a criterion based on the measurement forecasts. Hence, the state noise variance and the measurement variance can be selected through sensitivity analysis of the measurement forecasts.

4.4 Adaptive Kalman Filtering

In the previous sections, the state noise variance and the measurement variance are treated as fixed quantities that have to be selected before applying the Kalman filter. In other words, there is no feedback from the measurements to update these variances. Adaptive Kalman filtering designates the family of Kalman filter schemes that contain this kind of feedback. Many of these adaptive methods are based on adjusting the state noise variance and the measurement variance to fulfill a Kalman filter theoretical assumption that the forecast residuals are uncorrelated. Examples of adaptive Kalman filter methods can be seen in (Mehra, 1970, 1972), (Brewer, 1976) and (Todini, 1976). Judging from reviewing these works and from the experience gained through testing the method of Kitanidis and Bras (1978) on the flow updating model presented in Chap. 5, several drawbacks are observed:
1. Many of these methods are based on strict conditions difficult to meet in practice. For example, the model suggested by Mehra (1970) requires a fixed value for the matrix $H$ in Eq. 4.3. Therefore, this model cannot be applied to many applications, even for the simple example presented in this chapter.

2. Most of these methods require the additional estimation of new parameters. One typical example is the work by Kitanidis and Bras (1978). In their model, the state noise variance is treated as a stochastic variable which requires the knowledge of its own variance. So, the problem of selecting the state noise variance for the non-adaptive Kalman filter has been replaced by the problem of selecting a variance for the state noise variance. Not only that the adaptive model is more difficult than the non-adaptive one, but the dramatic increase in the computational cost does not materialize in practical improvement of the model performance.

3. The benefits from these methods are quite questionable, since they usually update the variances within small ranges, and therefore result in no significant changes in the performance of the updating models.

4. The application of these adaptive methods usually requires more measurements than the standard Kalman filter, which is a serious problem for most hydrological applications.

Although adaptive Kalman filtering is not recommended, the final decision should be based on comparative studies between adaptive and nonadaptive options.

4.5 **Extended Kalman Filtering**

As mentioned before, the Kalman filter can only be applied to linear systems, However this problem may be bypassed by linearizing nonlinear model equations using a Taylor
expansion (Gelb, 1986). But this approach is not suitable for threshold-type relationships which are very common in hydrological applications. These highly nonlinear threshold-type relationships can be linearized using the method of statistical linearization (Gelb, 1986). Examples of extended Kalman filtering applications can be found in the works of Kitanidis and Bras (1980c) and Georgakakos (1986).

4.6 Summary and Conclusion

This chapter starts with presenting a simple application of the Kalman filter in updating flow forecasts. The algorithm of this application is presented in the general state-space format suitable for the Kalman filter.

The Kalman filter is shown to be a Bayesian approach in which a priori normal probability density function is updated by a new measurement to yield a posterior probability density function.

The state noise and the measurement variances play a key role in controlling the updating process in the Kalman filter. The degree of updating is a function of relative values of these variances. A large state noise variance and a small measurement variance result in significant updating of the states; while a small state noise variance and large measurement variance result in poor updating. These variances should be treated as design parameters that are selected such that there is a balance between the model's ability to track changes in the states, and yet being insensitive to random noises. Three main factors should be considered in the process of these variances: the variability of the state, the quality of the measurement and the forecast lead. Relatively large state noise variance and small measurement variance are recommended for a highly variable system, good measurements and long forecast leads; while small state noise variance and large measurement variance are more suitable to the opposite situation. Very large state noise
variances should be avoided, unless the measurements are of a very high quality.

The adaptive Kalman filter methods that have been reviewed are found unsuitable for practical applications. This is mainly due to the strong conditions required for their application, their complex structures and their inconsistent performance. However, comparative studies should be carried on before making a final decision on using them.

Although the Kalman filter can only be applied to linear systems, the problem can be solved by linearizing nonlinear equations. Taylor expansion is a very common linearizing technique, but is not suitable for the threshold-type relationships which must be linearized using the statistical linearization technique.
Chapter 5

A Flow Updating Model

Most of the conceptual flow forecasting models are highly complex and nonlinear making it difficult to adjust the structures of these models for real-time operation. As has been mentioned in Chap. 3 error prediction and flow regression approaches offer simple and practical solutions to this problem. In this chapter a flow regression-type model is presented, in which on-line flow measurements are used to update flow forecasts generated from the conceptual UBC watershed model. The updating process is achieved using the Kalman filter.

A general description of the UBC watershed model with emphasis on the features relevant to the flow updating model is presented in section 5.1. This is followed by presenting the main equation of the flow updating model (section 5.2). The state-space format and the algorithm of the flow updating model are presented in sections 5.3 and 5.4, respectively. Methods of selecting the state noise variance and the measurement variance for applying the updating model on data from the Illecillewaet basin in British Columbia are discussed in section 5.5. The results of this application are presented and discussed in section 5.6.

5.1 The UBC Watershed Model

The UBC watershed model was designed by Quick and Pipes (1989b) to forecast streamflows from mountainous watershed systems mainly driven by snowmelt. The model requires daily inputs of maximum and minimum temperatures and precipitation, which are
usually selected as historical data of an average hydrological year. The model calculates on a daily basis the time distribution of the streamflows leaving the catchment as a result of snowmelt and rainfall.

A general flow chart of the UBC model is shown in Fig. 5.1. The average temperature is used to estimate the ratio of the snowfall to the rainfall in total precipitation. Melting of the snowpack is simulated by a heat transfer model using data on the maximum and the minimum temperatures\(^1\). The snowmelt and the rainfall are then distributed according to a soil moisture deficit model into evapotranspiration losses and inputs to watershed routing functions to generate runoffs at the basin outlet. These runoffs can be classified according to their origins into the following three flow components:

- the snowmelt flow component \(Q_s\) generated from fast routing of the snowmelt;
- the rain flow component \(Q_r\) generated from fast routing of the rainfall; and
- the groundwater flow component \(Q_g\) generated from slow routing of both the rainfall and the snowmelt.

Finally, the flow at the basin outlet, \(Q_{ubc}\), is estimated as the sum of these three flow components, i.e.:

\[
Q_{ubc}(t) = Q_s(t) + Q_r(t) + Q_g(t)
\]  

(5.1)

5.2 Modifying the UBC Watershed Model

The flow updating model is developed based on modifying Eq. 5.1 by estimating the flow as a weighted sum of the three flow components \(Q_s\), \(Q_r\) and \(Q_g\), i.e.:

\[
Q_s(t) = \alpha_s(t)Q_s(t) + \alpha_r(t)Q_r(t) + \alpha_g(t)Q_g(t)
\]  

(5.2)

\(^1\)This snowmelt model is presented in detail in Chap. 6.
Fig. 5.1 A general flow chart of the UBC watershed model.
where \( Q_e \) is the estimated flow; and \( \alpha_s, \alpha_r, \) and \( \alpha_g, \) are weighting factors for the flow components \( Q_s, Q_r \) and \( Q_g, \) respectively. These weighting factors are assumed to be stochastic and normally distributed variables that are related to their past values as follows:

\[
\begin{align*}
\alpha_s(t_k) &= \alpha_s(t_{k-1}) + w_s(t_{k-1}) \\
\alpha_r(t_k) &= \alpha_r(t_{k-1}) + w_r(t_{k-1}) \\
\alpha_g(t_k) &= \alpha_g(t_{k-1}) + w_g(t_{k-1})
\end{align*}
\]

where \( t_k - t_{k-1} = \Delta t \) is the forecast lead; \( w_s, w_r, \) and \( w_g \) are random errors (or white noise processes) of zero means and variances \( \sigma^2_{ws}, \sigma^2_{wr}, \) and \( \sigma^2_{wg}, \) respectively.

The Kalman filter is used to update the estimates of the means and the variances of the weighting factors whenever a flow measurement is available. The mean values serve as the best estimates of these factors; while the variances determine the confidence limits of these estimates.

The above flow updating model is based on the assumption that a forecast residual from the UBC watershed model is linearly related to the corresponding flow components. The validity of this assumption is very important for the updating model to be effective, especially for long forecast leads. This assumption has been proved to be valid for the data record of the Illecillewaet basin, British Columbia, using the least square criterion \( PLE, \) defined in Chap. 2.
5.3 The State-Space Format of the Flow Updating Model

Eqs. 5.2 and 5.3 are formulated into the following state-space model which is necessary for the application of the Kalman filter:

\[
\begin{align*}
X_k &= x_{k-1} + w_{k-1} \quad \text{State Eq.} \\
Z_k &= H_k x_k + v_k \quad \text{Measurement Eq.}
\end{align*}
\]  

(5.4)

where

\[
X_k = \begin{bmatrix}
\alpha_s(t_k) \\
\alpha_r(t_k) \\
\alpha_g(t_k)
\end{bmatrix};

w_k = \begin{bmatrix}
w_s(t_k) \\
w_r(t_k) \\
w_g(t_k)
\end{bmatrix};

z_k = [Q_o(t_k)], \text{where } Q_o \text{ is the observed flow};

H_k^T = \begin{bmatrix}
Q_s(t_k) \\
Q_r(t_k) \\
Q_g(t_k)
\end{bmatrix} \quad \text{and } v_k = [v(t_k)]
\]

where \( v \) is the measurement noise of zero mean and a known variance \( \sigma_v^2 \).

5.4 Model Algorithm

Let,

\[
G_k = \begin{bmatrix}
\sigma_{ws}^2(t_k) & 0.0 & 0.0 \\
0.0 & \sigma_{wr}^2(t_k) & 0.0 \\
0.0 & 0.0 & \sigma_{wg}^2(t_k)
\end{bmatrix};

P_k = \begin{bmatrix}
\sigma_{zs}^2(t_k) & \sigma_{sr}^2(t_k) & \sigma_{sg}^2(t_k) \\
\sigma_{sr}^2(t_k) & \sigma_{zs}^2(t_k) & \sigma_{rg}^2(t_k) \\
\sigma_{sg}^2(t_k) & \sigma_{rg}^2(t_k) & \sigma_{wg}^2(t_k)
\end{bmatrix} \quad \text{and}

R_k = [\sigma_o^2(t_k)]. \text{Diagonal elements of } P \text{ are the state variances and the nondiagonal elements are the covariances between pairs of the weighting factors. For example, } \sigma_{zs}^2 \text{ is the variance of the state } \alpha_s, \text{ and } \sigma_{sr}^2 \text{ is the covariance between the snowmelt and the rain weighting factors, } \alpha_s \text{ and } \alpha_r, \text{ respectively.}
In the off-line UBC watershed model the weighting factors are implicitly given fixed values of unity. Therefore, in the flow updating model the initial values of the weighting factors are set equal to unity, i.e.:

$$\hat{x}_0 = \begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \end{bmatrix}$$

The initial value of each weighting factor variance is selected equal to the square of 10% of the corresponding weighting factor; while the initial values of the covariances are selected equal to zero, i.e.:

$$P_0 = \begin{bmatrix} 0.01 & 0.0 & 0.0 \\ 0.0 & 0.01 & 0.0 \\ 0.0 & 0.0 & 0.01 \end{bmatrix}$$

The algorithm of the flow updating model is the same as the one presented in section 4.1.3. The algorithm is presented in details and supplemented with sample calculations in Appendix B.

### 5.5 Selection of the State Noise Variance and the Measurement Variance

From the discussion in Chap. 4 it has been shown that the Kalman filter is mainly controlled by the state noise variance and the measurement variance. Best model performance can be therefore achieved through selecting optimal values of these variances. Since it is the relative not the absolute values of these variances that influence the updating process\(^2\), it is sufficient to optimize model performance, through sensitivity analysis, versus either the state noise variance or the measurement variance while keeping the other variance fixed. In the application presented in this chapter the measurement variance

\(^2\text{This has been proven in section } 4.3.\)
was selected as a variable for the sensitivity analysis, while the state noise variance was kept fixed.

The flow updating model is applied to the flow forecasts of the Illecillewaet river, a tributary to the upper Columbia river, British Columbia. The streamflow is mainly supplied by snowmelt from a 10,000 square mile-basin\(^3\). Highest flows are observed during the warm spring and summer seasons. For this reason the period from April 15 to July 15, 1970 has been selected for testing the model performance. This period is also characterized by rapidly changing flows that are difficult but yet important to forecast.

Evaluating model performance by the coefficient of efficiency, \(E\), defined in Chap. 2, sensitivity analysis can be applied for two types of measurement variances: a fixed value during the evaluation period (Figs. 5.2 and 5.3); or as the square of a fixed percentage of the observed flow (Fig. 5.4). Sensitivity analysis, for each type of the measurement variance, is performed for several forecast leads (1, 5, 10, 20 and 30 days). The state noise variances have been fixed at values equal to 0.01, calculated as the square of 10\% of the initial values of the corresponding weighting factors, which are equal to unity, i.e:

\[
G_0 = \begin{bmatrix}
0.01 & 0.00 & 0.00 \\
0.00 & 0.01 & 0.00 \\
0.00 & 0.00 & 0.01 \\
\end{bmatrix};
\]

5.5.1 Fixed Measurement Variance

Fig. 5.2 shows that the criterion \(E\) is relatively insensitive to fixed measurement in the range of 0 to 1000 \(\left(\frac{m^3}{s}\right)^2\), with the exception of the forecast lead of 20 days where the model performance is noticed to be very low for very small measurement variances (near 0 \(\left(\frac{m^3}{s}\right)^2\)). For significantly larger measurement variances (up to 20,000 \(\left(\frac{m^3}{s}\right)^2\)), Fig. 5.3 shows that model performance deteriorates with increasing measurement variance with

\(^3\)See the description of the hydrology and the climate of this basin in Appendix A.
Fig. 5.2 The coefficient of efficiency, $E$, vs. fixed measurement variance (0 to 1000 $\left( \text{m}^3 \text{s}^{-2} \right)$).
Fig. 5.3 The coefficient of efficiency, $E$, vs. fixed measurement variance ($0$ to $20,000 \left( \frac{m^3}{s} \right)^2$).
Fig. 5.4 The coefficient of efficiency, $E$, vs. percentage measurement variance.
rates dependent on the forecast lead. Short-term forecasts (especially for one day) are minimally affected by increasing the measurement variance; while long-term forecasts (in particular the 25 and 30 days) deteriorate to levels closer to the off-line forecasts for large measurement variances. The effect of the measurement variance on the updating process and the frequency of updating explain these results. Large measurement variance makes the updating model less "responsive" to new measurements which results in less updating of the weighting factors. The availability of a large number of measurements for short forecast leads (92 measurements for a forecast lead of one day) counterbalance the effect of large measurement variance by increasing the number of times the weighting factors are updated. In contrast, the weighting factors are updated much less frequently (3 times for forecast lead of 30 days) for long forecast leads resulting in minimal updating of the forecasts. Although these observations suggest, in general, that measurement variance is to be selected on the smaller side for long forecast leads, very small measurement variances should be avoided, since a very small measurement variance causes the updating model to be very sensitive to errors in the measurements\(^4\). Therefore, a poorly measured flow may cause the model to erroneously alter the weighting factors. The effect of this "bad" measurement is particularly harmful for long-term forecasts since a small number of future measurements are available to bring the weighting factors back on the "correct" course. This explains the inferior flow forecasts observed for forecast lead of 20 days when very small measurement variances are used (see Fig. 5.2.)

Since the effect of the measurement variance is the opposite of that of the state noise variance, i.e. the effect of a large measurement variance is similar to that of a small state noise variance and vice versa, the discussion above is equivalent to that in section 4.3.

\(^4\)A measurement variance of zero causes weighting factors to be updated solely based on the measurements, i.e., the updating model completely "trust" the measurements.
5.5.2 Percentage Measurement Variance

Measurement variance can be selected as the square of a certain percentage of the flow. This reflects what has been reported by many researchers that high flow measurement errors are associated with higher flows (see Quick 1990.) Fig. 5.4 shows plots of $E$ vs. measurement variance as a square of percentage flow ranging from 0% to 100% for forecast leads of 1, 5, 10, 20 and 30 days. It is interesting to notice the similarity between Fig. 5.4 and Fig. 5.3. However, the peaks of the percentage variance curves are flatter and start later than those of fixed variance curves. This is because the percentage axis in Fig. 5.4 is not linear in terms of $(m^3_s)^2$ units. For example, for a flow of 100 $(m^3_s)$ a 20% variance equals $[(0.2)(100)]^2 = 400 (m^3_s)^2$; while a 40% variance equals $[(0.4)(100)]^2 = 1600 (m^3_s)^2$.

Fig. 5.4 shows that the updating model attains its highest levels of performance for measurement variances between 10% to 20%. Comparing Figs. 5.3 and 5.4 shows no practical differences between selecting the measurement as a fixed value or as the square of a percentage of the flow. This is only true based on the average performance of the updating model as measured by the coefficient of efficiency. As will be demonstrated later in this chapter a more detailed look at the model performance for both types of variances shows an important difference between the two approaches.

5.5.3 An Adaptive Kalman Approach

An adaptive Kalman filter model developed by Kitanidis and Bras (1978b) has been tested on the updating model. No significant differences have been noticed between the adaptive and the non-adaptive approaches. This however was not unexpected, since the adaptive algorithm is designed for slow time-varying variances resulting in relatively insignificant changes in the variances. It is also doubtful that these changes in the variances will
always improve flow forecasts. In addition the adaptive model is quite complex, time consuming and requires the knowledge of additional parameters. In general the adaptive approach is not recommended for the current flow updating model or similar models (see Chap. 4.)

5.6 Application

Fig. 5.2 shows that the highest levels of performance of the updating model are those for measurement variances between 200 \( \left( \frac{m^3}{s} \right)^2 \) to 500 \( \left( \frac{m^3}{s} \right)^2 \). Therefore, a measurement variance of 200 \( \left( \frac{m^3}{s} \right)^2 \) is selected to update flow forecasts.

5.6.1 Overall Performance

The observed flows, the off-line flow forecasts and flow forecasts for forecast leads of 1, 10, 20 and 30 days and their corresponding residuals are shown in Figs. 5.5, 5.6, 5.7 and 5.8, respectively. Updating flow forecasts on a daily basis resulted in much better correspondence between measured and calculated hydrographs (see Fig. 5.5.) This is reflected in the great improvement in the value of \( E \) from 0.74 to 0.95 (a 27% increase) (see Table 5.1.) The updating model performs even better than the powerful but naive short-term persistence and extrapolation models by 31% and 40%, respectively (see Table 5.1.). This is rarely observed in the real-time forecasting literature (see Chap. 3.) Accumulated errors were also dramatically reduced from 1672 to 38 \( \frac{m^3}{s} \).

Updating flow forecasts every 10 days did not result in any change in the flow forecasts for the first 40 days (see the period from April 15 to May 15 in Fig. 5.6,) since only 3 measurements were available and forecast residuals were small. However, flow forecasts were greatly improved for the following period. This resulted in a remarkably high \( E \)
Fig. 5.5 Hydrographs of the observed, UBC and updated flows and residuals for a forecast lead of one day.
Fig. 5.6 Hydrographs of the observed, UBC and updated flows and residuals for a forecast lead of 10 days.
Fig. 5.7 Hydrographs of the observed, UBC and updated flows and residuals for a forecast lead of 20 days.
Fig. 5.8 Hydrographs of the observed, UBC and updated flows and residuals for a forecast lead of 30 days.

Dates of Measurements Used in Updating (1970)
Chapter 5. A Flow Updating Model

Table 5.1 Evaluation of the flow forecasts in terms of several criteria.

<table>
<thead>
<tr>
<th>Model</th>
<th>Coefficient of Efficiency</th>
<th>Coefficient of Determination</th>
<th>Coefficient of Persistence</th>
<th>Coefficient of Extrapolation</th>
<th>Accumulated Error m³/s</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>E</td>
<td>D</td>
<td>Cₚ</td>
<td>Cₑ</td>
<td></td>
</tr>
<tr>
<td>Q₂UBC</td>
<td>0.74</td>
<td>0.94</td>
<td>-2.26, Δt = 1</td>
<td>-1.83, Δt = 1</td>
<td>1672</td>
</tr>
<tr>
<td>Q₂U₁</td>
<td>0.95</td>
<td>0.95</td>
<td>0.62, Δt = 10</td>
<td>0.97, Δt = 10</td>
<td>38</td>
</tr>
<tr>
<td>Q₂U₁₀</td>
<td>0.93</td>
<td>0.93</td>
<td>0.76, Δt = 20</td>
<td>0.99, Δt = 20</td>
<td>-389</td>
</tr>
<tr>
<td>Q₂U₃₀</td>
<td>0.92</td>
<td>0.92</td>
<td>0.82, Δt = 30</td>
<td>1.00, Δt = 30</td>
<td>-360</td>
</tr>
<tr>
<td>Q₂U₃₀</td>
<td>0.81</td>
<td>0.88</td>
<td>0.82, Δt = 30</td>
<td>1.00, Δt = 30</td>
<td>470</td>
</tr>
</tbody>
</table>

value of 0.93. The updating model did also better than the persistence and extrapolation models. This is expected since the persistence and extrapolation models are solely dependent on the correlations between consecutive flows which are very weak for long forecast leads. Accumulated errors were reduced down to -389 m³/s.

Updating forecast leads every 20 days (Fig. 5.7) resulted in improved forecasts after May 25. A High E value of 0.93 has been achieved. The ability of the updating model to greatly improve flow forecasts even for this relatively long forecast lead is attributed to the strength of the underlying off-line UBC watershed model and the validity of the assumption that the flow forecast residuals of the UBC watershed model are highly linear. It should be emphasized that the role of the flow updating model is to remove linear errors from the flow forecasts of the UBC model, therefore, if the UBC model is not representative of the watershed system, the flow forecasts are not expected to improve
by using the flow updating model, especially for long forecast leads.

Updating flow forecasts every 30 days showed improvement only after June 14. $E$ increased to 0.81 and accumulated errors were reduced down to $470 \text{ m}^3$. This is an important improvement, since long-term volume forecasts are important for reservoir operation.

5.6.2 Examining Progressive Model Performance

Fig. 5.9 shows successive daily updating of the flow forecasts for forecast leads from one to 10 days for the period starting May 25 and ending June 15, 1970. Observed hydrograph, updated forecasts and residuals are shown in the unshaded area. The shaded area to the left contains past observed flows; while the shaded area to the right shows future observed flows not covered by the updating model. Measurement variance has been selected equal to a fixed value of 200 $\left(\text{m}^3\right)^2$. Fig. 5.10a shows the computed flow components $Q_s, Q_r$, and $Q_q$ for the same period. Fig. 5.10b shows the daily updated estimates of the weighting factors that were used to generate the flow forecasts shown in Fig. 5.9a.

As a result of underestimating the flow on May 26 (see Fig. 5.9a), all the weighting factors were increased (see Fig. 5.10b) to compensate for this negative flow forecast residual. Using these weighting factors for updating flow forecasts for the next ten days resulted in overestimating low flows and improving high flow forecasts, especially at the hydrograph peak on June 4 (see Fig. 5.9b.). The flow forecast residual on May 27 was quite small leading to insignificant change in the weighting factors (see Fig. 5.10b.). On May 28 positive flow forecast residual was encountered. The updating model responded by decreasing the groundwater weighting factor and, to less extent, the rain weighting factor. However, the snowmelt weighting factor was increased. This is due to the highly negative correlation between the groundwater and the snowmelt weighting.
Fig. 5.9(a,b,c,d,e&f) Successive daily updating of flow forecasts for forecast leads of 1,2,...,10 days (fixed $R = 200 \text{ m}^3\text{s}^{-1}$). See key in Fig. 5.12.
Fig. 5.9(g,h,i,j,k&l) Successive daily updating of flow forecasts for forecast leads of 1, 2,..., 10 days (fixed $R = 200 \left( \frac{m^3}{s} \right)$). See key in Fig. 5.12.
Chapter 5. A Flow Updating Model

Fig. 5.10a The calculated flow components.

Fig. 5.10b Weighting factors (fixed $R = 200 \left( m^3 / s \right)$).

Fig. 5.10c Weighting factors ($R = (15\% \text{ Flow})^2$).
factors and the dominance of the groundwater flow component\(^5\) during this period (see Fig. 5.10a.) The reduction in the groundwater weighting factor resulted in improving low flow forecasts; while the increase in the snowmelt weighting factor improved high flow forecasts, particularly the peak (see Fig. 5.9d.) The trend in decreasing the contribution of the groundwater\(^6\) and increasing that of the snowmelt continued on May 29 (Fig. 5.9e) resulting in more improvement in flow forecasts.

The flow forecasts updated on June 2 (two days before the peak) (see Fig. 5.9i) show a significant improvement and a very close estimation of the maximum flow. This is an important result, since most real-time flow forecasting models experience serious problems on the rising limb (see Chap. 3.) However, the high negative residual observed on the next day, June 3, (Fig. 5.9i) caused the snowmelt weighting factor to be increased dramatically to reach a value close to unity (see the peak in Fig. 5.10b,) resulting in overestimated flow forecasts, especially for the peak (Fig. 5.9j). This deterioration in model performance is mainly due to the high variability of flows on the rising limb complicated by the difficulty in accurately measuring high flows (see Quick (1990).) This problem can be successfully controlled by setting the measurement variance as a percentage of the flow, causing the updating model to be less sensitive to measurement errors, or rapid changes in high flows; while being able to extract useful information from the more accurate low flows. Fig. 5.11 is similar to Fig. 5.9 except that the measurement variance is selected as the square of 15\% of the flow. A significant improvement in the flow forecasts is noticed for the hydrograph of Figs. 5.11i and 5.11j, especially the hydrograph peak on June 3, (compare with Figs. 5.9i and 5.9j) without affecting, if not improving, the performance of the updating model on other days. Selecting a percentage-wise measurement caused more control of the changes in the weighting factors as can be seen from comparing Fig.

---

\(^5\)This is explained in detail in Appendix B.

\(^6\)The rain component is insignificant as can be seen from Fig. 5.10a.
Fig. 5.11 (a,b,c,d,e&f) Successive daily updating of flow forecasts for forecast leads of 1, 2, ..., 10 days ($R = (15\% \text{ flow})^2$). See key in Fig. 5.12.
Fig. 5.11 (g,h,i,j,k,l) Successive daily updating of flow forecasts for forecast leads of 1, 2, ..., 10 days (R = (15% flow)). See key in Fig. 5.12.
Fig. 5.12 The key figure for Figs. 5.9 and 5.11.
5.10c with Fig. 5.10b. Particularly, the peak in the snowmelt weighting factor (on June 3) is reduced. These results support the approach of selecting the measurement variance as a percentage of the flow rather than a fixed value.

It is interesting to notice that model performance is best on the hydrograph falling limb as can be seen from plots (k) and (i) of Figs. 5.9 and 5.11.

5.6.3 Detection of Linearity in The Flow Forecasts Residuals

The flow updating model presented in this chapter is based on the assumption that residuals of the flow forecasts generated by UBC Watershed model are linearly related to the prefinal outputs, namely: the flow components $Q_s$, $Q_r$ and $Q_g$. This assumption can be satisfactorily tested by calculating the least squares criteria $PLE$ defined in Chap. 2. A high value of $PLE$ indicates a highly linear residuals, and vice versa.

Since the Kalman filter is based on minimizing the sum of squares of errors, $PLE$ could be used to test the possibility of improving flow forecasts using the Kalman filter. In the present work $PLE$ is calculated from the $E$ value of the UBC Watershed model and $D$ from linear regression of $Q_o$ on $Q_s$, $Q_r$ and $Q_g$. $E$ is equal to 0.74, while $D$ is equal to 0.94, so

$$PLE = \left( \frac{0.94 - 0.74}{1 - 0.74} \right) (100) = 77\%$$

This means that 77% of the errors are linearly related to $Q_s$, $Q_r$ and $Q_g$ and there is a great potential of improving flow forecast using the flow updating model. This is shown in the improvement of $E$ value from 0.74 to 0.95 (for forecast lead of one day.) That results in removing $\frac{0.95 - 0.74}{1 - 0.74} = 81\%$ of the errors.

The $PLE$ can be used in the process of selecting the optimal state-space structure of the flow updating model, avoiding therefore the time consuming and expensive process of implementing each model. This can be done by calculating the $PLE$ values of each
Chapter 5. A Flow Updating Model

model and then selecting the model with the highest $PLE$ value.

5.6.4 Summary and Conclusion

A flow forecast updating model is designed based on the assumption that the residuals of the flow forecasts generated by the UBC watershed model are linearly related to the calculated flow components $Q_a$, $Q_r$ and $Q_g$. The flow is estimated as a weighted sum of these flow components. Feedback of flow measurements are used to update the corresponding weighting factors using the Kalman filter technique.

Sensitivity analysis based on data from the Illecillewaet basin shows that optimal measurement variances range between 200 to 500 $(\text{m}^3 \text{s}^{-2})$ for fixed variances, and between 10% to 20% for percentage variances. These values are obtained for state noise variances equal to 0.01. Using a scheme developed by Kitanidis and Bras (1978) for updating state noise variance results in no improvement to flow forecasts.

The flow updating model has been applied on the Illecillewaet basin for forecast leads of 1, 10, 20 and 30 days. A significant improvement in flow forecasts has been observed even for a long forecast lead of 20 days. The high performance of the flow updating model for long forecast leads is due to two reasons: the ability of the underlying UBC watershed model to simulate the Illecillewaet basin, and the validity of the assumption that flow forecast residuals are linear. The assumption of linearity is tested using the least square criterion $PLE$.

Although average performance of the flow updating model is not affected by the type of measurement variance, selecting the measurement variance as a square percentage of the flow improves flow forecasts in the vicinity of high flows. This can be attributed to the observation by many researchers that the quality of flow measurements deteriorates as the flow increases, therefore selecting the measurement variance as a square percentage of the flow reduces sensitivity of the flow updating model to the high errors in the high
flows.

The flow updating model provides an effective and flexible mean for using feedback of flow measurements in improving flow forecasts of a highly complex watershed model. It can also be used in conjunction with the snowpack model presented in Chap. 6 to further enhance the flow forecasts of the UBC watershed model.
Chapter 6

A Snowpack Updating Model

In mountainous watershed systems like those of the Columbia, Peace and Fraser rivers in British Columbia and those feeding the Indus River system, snowmelt constitutes the major input to streamflows during the warm spring and summer seasons, which are characterized by the most extreme streamflows. Forecasting seasonal streamflows therefore depends on accurate assessment of the snowpack water equivalents stored in the watershed. The most effective method of updating streamflows forecasts is therefore likely to depend on an updating procedure for correcting these snowpack estimates.

In this study a model has been developed to update parameters of an energy budget snowpack model (section 6.2) using real-time snowpack measurements. This energy budget snowpack model is a sub-model of the UBC watershed model. The updating procedure is based on calculating the value of a snowpack parameter that yields a perfect correspondence between measured and calculated snowpacks. The updated snowpack parameter is then used in calculating future snowpack values and streamflows forecasts. The updating procedure is repeated whenever a new snowpack measurement becomes available.

Updating a snowpack parameter from snowpack measurements necessitates that this parameter is expressed as a function of two consecutive snowpack measurements and the accumulated snow fallen during the time period between these measurements. If the snowpack is measured on a daily basis (or, in general, on the same intervals as the flow forecasts), then the snowpack updating function is linear and can be easily
formulated. However, if snowpack measurements are sparse (e.g., available on a monthly basis), then the development of the snowpack has to be tracked on a daily basis from the last snowpack measurement back to the previous one. Much more effort is therefore needed to develop the snowpack updating function. The resulting function can still be linear as long as there are no changes in the thermal condition of the snowpack (i.e., from frozen to melting condition or vice versa) during the interval between two consecutive snowpack measurements. Since thermal condition of the snowpack may vary, especially as the snowpack reaches its peak, a more comprehensive snowpack updating function has been developed. First, the snowpack updating function has been formulated separately for each of the three possible thermal conditions of the snowpack (which are the frozen, partially melting and the completely melting conditions); then the final updating function is expressed as the sum of these three functions each multiplied by a factor (named tag) that is either equal to unity if the snowpack is in the same condition or equal to zero if the snowpack is not in the same condition. An iterative method is used to determine the correct values of these tags, where the tags are first set equal to the initial or the previous values, then new snowpack values are calculated using these tags. New values of the tags are then determined and this iterative procedure is repeated until the last set of tags values matches that of the previous iterative step.

This chapter starts with discussing the snowpack energy budget approach and the snowpack routine of the UBC watershed model which was developed based on this approach. Then, the results of studying the effects of the cloud cover, the snow albedo and the snowfall on the snowpack are presented and discussed. The approach of developing the snowpack updating model and its algorithm are then presented. Finally, the results from the application of the snowpack updating model on the Mount Fidelity data station and the Illecillewaet basin in British Columbia are presented and discussed.
6.1 The Energy Budget of the Snowpack

The snowpack can be treated as a thermodynamic system which receives and/or loses heat through shortwave, longwave, conductive, advective and rain melt heat fluxes. The loss in the snowpack is calculated as the snowmelt, estimated equal to the water column melted by the heat transfer surplus to the snowpack, less the snowfall. The heat surplus to the snowpack can be calculated as follows: equation:

\[ H_t = H_s + H_l + H_c + H_a + H_r \]  

(6.1)

where \( H_t \) is the sum of heat fluxes;

\( H_s \) is the shortwave heat component;

\( H_l \) is the longwave heat component;

\( H_c \) is the convective heat component;

\( H_a \) is the advective heat component; and

\( H_r \) is the advective heat transfer by rain.

The snowmelt, \( m \), is determined according to \( H_t \) and the water equivalent of the snowpack, \( d_m \), as follows:

\[
\begin{align*}
\text{For } H_t < 0 & \quad m = 0.0 \\
\text{For } H_t > 0 & \quad m = \begin{cases} 
H_t & \text{if } d_m > H_t \\
 d_m & \text{if } d_m < H_t
\end{cases}
\end{align*}
\]

The shortwave heat component is a function of the snow albedo and the cloud cover (section 6.1.1); while the longwave heat component is a function of the cloud cover and the temperature (section 6.1.2). The convective and the advective heat components are
functions of the wind speed and the temperature (section 6.1.3). In general, the snow albedo, the cloud cover and the wind speed are rarely measured, and so, these quantities are estimated as functions of the temperature (Quick and Pipes, 1989a). The parameters of these functions are usually treated as fixed quantities. However, in the snowpack updating model, to be presented later in detail, the most sensitive of these parameters are considered to be dynamic quantities that are updated as functions of the snowpack measurements.

6.1.1 The Shortwave Heat Influx

Shortwave heat influx can be expressed as follows (Quick and Pipes, 1989a):

\[ H_s(t) = I_s(t) [1 - C(t)] [1 - A(t)] \text{ in mm/day} \quad (6.2) \]

where \( C \) is the fraction of cloud cover; \( A \) is the snow albedo; and \( I_s \) is the incident solar radiation for 35° North estimated as follows:

\[ I_s(t) = 54 - 29 \cos(2\pi N/365) \text{ in mm/day} \quad (6.3) \]

where \( N \) is the number of days starting from the first of January.

6.1.2 The Longwave Radiation

The longwave radiation of a melting snowpack can be expressed as follows (Quick and Pipes, 1989a):

\[ H_l(t) = [-20 + 0.94 T_a(t)] [1 - C(t)] + 1.24 T_c(t) C(t) \quad (6.4) \]

where \( T_a \) is the average temperature in °C; and \( T_c \) is the cloud temperature (assumed equal to the minimum temperature).
6.1.3 The Convective and Advective Heat Transfer

The convective heat transfer is estimated as follows (Quick and Pipes, 1989a):

\[ H_c = 0.113 R_m (P/101) T_a(t) v_w \text{ in mm/day} \]  \hspace{1cm} (6.5)

where \( P \) is the atmospheric pressure in \( kN/M^2 \) for the elevation under consideration; \( v_w \) is the wind speed in kilometers per hour; and \( R_m = 1 - 7.7 R_i \), where \( R_i \) is the bulk Richardson number that can be approximated as \( 0.095 T_m(t) / v_w^2 \).

The advective heat transfer can be expressed as follows (Quick and Pipes, 1989a):

\[ H_a = 0.44 R_m T_d v_w \]  \hspace{1cm} (6.6)

where \( T_d \) is the dew point, which can be approximated by the minimum air temperature.

6.1.4 The Rain Melt

Rain melt, \( H_r \), can be estimated as follows (Quick and Pipes, 1989a):

\[ H_r = K T_n p_r \text{ in mm/day} \]  \hspace{1cm} (6.7)

where \( K \) is a constant set equal to 0.01249; and \( T_n \) is the minimum air temperature.

6.2 Calculating the Snowpack in the UBC Watershed Model

In the UBC watershed model the snowpack is calculated according to a routine based on the energy budget approach presented in the previous section. The algorithm of this routine is presented below and its flow diagram is shown in Fig. 6.1:

1. The average temperature is calculated as the mean of the measured maximum, \( T_x \), and minimum, \( T_n \), temperatures, i.e:

\[ T_a(t) = \frac{T_x(t) + T_n(t)}{2} \]  \hspace{1cm} (6.8)
Chapter 6. A Snowpack Updating Model

Fig. 6.1 Flow chart of the UBC snowpack model.
2. The snowfall, $p_s$, is estimated from the total precipitation according to the routine presented in section 6.3.3.

3. The shortwave, the longwave, the convective, the advective and rain melt heat fluxes are calculated according to the equations in section 6.1.

4. The energy surplus from the snowpack, $H_e$, is calculated as the sum of $H_t$ (Eq. 6.1) and the previous cold content\(^1\) of the snowpack, $H_{cc}(t-1)$, after being reduced by a factor $\lambda$ to account for the warming of the frozen snowpack during the day, i.e:

$$H_e(t) = \lambda H_{cc}(t-1) + H_t(t)$$

where $0.0 < \lambda < 1.0$.

5. The snowmelt, $m(t)$, the water equivalent of the snowpack, $d_m(t)$, and the new cold content of the snowpack, $H_{cc}(t)$, are estimated as follows:

- For melting snowpack, $H_e(t) > 0.0$:

\[
\begin{align*}
    m(t) &= H_e(t) \\
    d_m(t) &= d_m(t-1) - m(t) + p_s(t) \quad \text{if } H_e(t) < d_m(t-1) + p_s(t) \\
    H_{cc}(t) &= 0.0
\end{align*}
\]

\[
\begin{align*}
    m(t) &= d_m(t-1) + p_s(t) \quad \text{if } H_e(t) > d_m(t-1) + p_s(t) \\
    d_m(t) &= 0.0 \\
    H_{cc}(t) &= 0.0
\end{align*}
\]

\(^1\)Cold content of the snowpack is defined as the amount of heat required to raise the temperature of the snowpack to $0.0^\circ C$, so the cold content is always less or equal to zero.
• For frozen snowpack, $H_e(t) < 0.0$:

\[ m(t) = 0.0; \quad d_m(t) = d_m(t - 1) + p_s(t); \quad \text{and} \quad H_{cc}(t) = H_e(t) \]

### 6.3 Sensitivity Analysis of the Snowpack Parameters

Since the changes in the snowpack are calculated according to the balance between the input of snowfall and the output of the snowmelt, the estimated snowpack is controlled by the parameters of the snowmelt and the snow formation functions. However these are only a few of these parameters to which the snowpack is significantly sensitive. Therefore, only these parameters are considered for the snowpack updating model. In the following sections four parameters are presented and their effects on the snowpack are studied and discussed.

#### 6.3.1 The Cloud Cover

The cloud cover affects snowmelt through its influence on the shortwave and the longwave heat transfer components (see Eqs. 6.2 and 6.4). In most cases cloud cover is not observed, especially in the remote but important regions of the mountainous watersheds. However, cloud cover can be estimated as a function of the daily temperature range as follows (Quick and Pipes, 1989a):

\[ C(t) = 1 - \frac{T_x(t) - T_n(t)}{\Delta T_s} \quad (6.9) \]

where $\Delta T_s$ is the daily temperature range for open sky conditions. Eq. 6.9 is modified to include a correction factor, $\eta_c$ as follows:

\[ C(t) = \eta_c \left[ 1 - \frac{T_x(t) - T_n(t)}{\Delta T_s} \right] \quad (6.10) \]

Snowpack sensitivity to the cloud cover can therefore be tested by comparing the snowpacks calculated using different values of the cloud cover correction factor $\eta_c$. Fig. 6.2
shows the snowpacks calculated for \( \eta_c \) values of 0.5, 0.75, 1.0, 1.25 and 1.50. It can be seen from this figure that increasing the value of \( \eta_c \), i.e. increasing the cloud cover, results in slightly slowing the rise in the snowpack during the accumulation period (when snowfall is larger than snowmelt) and significantly slowing the meltdown of the snowpack during the depletion period (when snowfall is less than the snowmelt).

The above observations are attributed to the differences in the conditions that prevail during the snowpack accumulation period and those of the snowpack depletion period. During the accumulation period the average daily temperatures are low (mostly below 0 °C) and snowfalls are intense. Low temperatures result in negative longwave, convective and advective heat transfers (see Eqs. 6.4, 6.5 and 6.6). Fresh snowfalls increase the snow albedos and thus reduce the shortwave heat component (see Eq. 6.2). These combined effects produce negative heat balance to the snowpack causing it to freeze. Increasing the cloud cover decreases the negative longwave component and may therefore lead to a minimal snowmelt that may be reduced by the decrease in the positive shortwave component. The total outcome is a slight reduction of the snowpack (which is difficult to observe in Fig. 6.2) due to the increase in the cloud cover. On the other hand, decreasing the cloud cover increases the negative longwave component and at the same time increases the shortwave component. The net result is a further decrease in the energy content of the snowpack, causing more freezing of the snowpack and no change in the snowpack.

During the snowpack depletion period average temperatures are relatively high and snowfall is much less than during the accumulation period. High temperatures result in positive longwave radiation, convective and advective heat transfers. Old snowcover has a small albedo value and therefore absorbs more shortwave energy. The net heat transfer to the snowpack is therefore positive resulting in snowmelt. Increasing cloud cover reduces

\[^2\text{Negative heat transfer indicates a loss of the snowpack energy content.}\]
Fig. 6.2 Sensitivity of the snowpack to the cloud cover parameter $\eta_c$. 
the positive shortwave and longwave components resulting in less snowmelt. This causes an increase in the snowpack and subsequently a delay in snowpack depletion. On the other hand, decreasing cloud cover increases the shortwave and longwave components resulting in more snowmelt that speeds snowpack depletion.

From the above discussion it can be concluded that an error in estimating the cloud cover causes a timing error in the snowpack that is much more pronounced during the snowpack depletion stage than during the accumulation stage (see Fig. 6.3.) So, for a given time during the depletion stage increasing cloud cover results in thicker snowpack (see the arrow on the right-hand side of Fig. 6.3;) while for another given time during the accumulation period increasing the cloud cover results in the opposite (see the arrow on the left-hand side of Fig. 6.3.)

Sensitivity of the snowpack to the cloud cover can be also tested using the parameter $\Delta T_s$ (see Eq. 6.9.) Taking $\Delta T_s = 20^\circ C$ as a standard value, the snowpack is calculated for $\Delta T_s$ values 25% and 50% higher than the standard (25 and 30 $^\circ C$, respectively) and 25% and 50% lower than the standard (15 and 10 $^\circ C$, respectively). It can be seen from Fig. 6.4 that increasing $\Delta T_s$ results in an increase in the total volume of the snowpack and an increase and a delay of the peak of the snowpack, which are attributed to the reduction of snowmelt during the depletion stage. A very small effect can be noticed during the accumulation stage. The effect of $\Delta T_s$ on the snowpack is similar to that of $\eta_c$ although it is stronger during the accumulation stage.

From the above discussion, it can be concluded that the snowpack is sensitive to the parameters $\eta_c$ and $\Delta T_s$, only during its depletion stage. Therefore, only measurements during that stage are valuable as feedback for updating these parameters.
Fig. 6.3 The effect of under- or over-estimating the cloud cover on the snowpack.
Fig. 6.4 Sensitivity of the snowpack to the cloud cover parameter $\Delta T_s$. 
6.3.2 The Snow Albedo

Albedo of a surface is defined as the ratio of the reflected shortwave to the incident shortwave energies. Fresh snow has a high albedo (0.90) that decreases exponentially with time to as low as 0.3. If the new snowfall is thick enough to completely cover the old snow, the albedo can rise back to its maximum value. Eq. 6.2 is modified as follows to include a correction factor, \( \eta_a \):

\[
H_s(t) = I_s(t)[1 - C(t)][1 - \eta_a A(t)] \quad \text{in mm/day} \tag{6.11}
\]

Fig. 6.5 shows the snowpacks calculated for five \( \eta_a \) values of 0.5, 0.75, 1.0, 1.25 and 1.50. From this figure it can be seen that increasing the snow albedo correction factor, \( \eta_a \), above the standard value of one results in delaying the melting and increasing the volume of the snowpack during the depletion stage without significantly affecting the snowpack during the accumulation stage. Reducing \( \eta_a \) below a value of unity results in a total reduction of the size of the snowpack. These observations can be attributed to the effect of snow albedo on the shortwave component. During the accumulation stage the snowpack is mostly frozen. Increasing the snow albedo results in a decrease in the shortwave component and therefore more freezing of the already frozen snowpack. Therefore, no change in the snowpack size is expected. Decreasing the snow albedo, on the other hand, increases the shortwave component which, if large enough, causes a snowmelt and a reduction in the snowpack.

During the snowpack depletion stage, when the snow is melting, increasing the snow albedo decreases the shortwave component and therefore reduces the snowmelt causing a delay in snow depletion. Consequently, decreasing the snow albedo increases the shortwave component leading therefore to more snowmelt.

In general, the effect of \( \eta_a \) on the snowpack is simpler than \( \eta_c \) and \( \Delta T_s \). This is because \( \eta_a \) only affects the shortwave energy component, while the other two parameters affect
Fig. 6.5 Sensitivity of the snowpack to the snow albedo parameter $\eta_a$. 
both the shortwave and longwave components.

It is interesting to notice from Eq. 6.2 that although the cloud cover and the snow albedo are two different physical phenomena, their effects on the shortwave heat transfer to the snowpack are similar. The cloud cover reduces the amount of the shortwave energy received by the snowpack by absorbing a portion of this energy, and the unabsorbed portion is later reduced by reflection from the snow surface. The interaction between these two processes makes it difficult to determine the source(s) of errors in the snowmelt estimates. For example, an error in the snowmelt that can be corrected by adjusting the parameter \( \eta_c \) may also be corrected through adjusting the parameter \( \eta_a \). The inability to determine the actual source(s) of errors in the snowmelt may lead to an erroneous adjustment of a snowpack parameter that may degrade future snowpack forecasts. The sources of errors in the snowmelt estimates, or the snowpack forecasts in general, may be best identified by studying the effect of synthesized errors in the snowpack parameters on the snowpack forecasts, as has been done in this section and sections 6.3.1 and 6.3.3.

6.3.3 The Snow Formation Function

In most precipitation gauges, the precipitation form (snow or rain) is unknown. Since the form of precipitation is important for the routing the water through the watershed, a simple routine is used in the UBC watershed model to estimate the ratio of snowfall, \( p_s \), to rainfall, \( p_r \), in the total precipitation (Quick and Pipes, 1989a):

- If the average temperature, \( T_a \), is less than 0.0°C, then \( p_s(t) = pp(t) \), where \( pp(t) \) is the precipitation at time \( t \).
- If \( T_a > T_r \), where \( T_r \) is the temperature above which all precipitation is rain, then \( p_s(t) = 0.0 \) and \( p_r(t) = pp(t) \).
Chapter 6. A Snowpack Updating Model

• If average temperature is greater than 0.0°C and less than $T_r$, then

$$p_s(t) = \left[ 1 - \frac{T_s(t)}{T_r} \right] pp(t)$$  \hspace{1cm} (6.12)

In the UBC watershed model $T_r$ is usually set equal to 2.0°C. In this study, however, $T_r$ is treated as a parameter that can be updated using snowpack measurements. Fig. 6.6 shows the snowpacks calculated for $T_r$ values of 1.0, 1.5, 2.0, 2.5 and 3.0 °C. As can be seen from this figure large $T_r$ values causes increases in the snowfalls leading to thicker snowpack. Small values of $T_r$ result in the opposite.

6.4 The Snowpack Updating Approach

In the UBC watershed model the snowpack parameters are selected through a calibration process using historical data and their values are fixed during the operation stage. In this study, however, the snowpack parameters, $\eta_c$, $\Delta T_s$, and $\eta_a$ are considered dynamic quantities that can be updated using on-line snowpack measurements. The general outline of the updating model has been presented at the beginning of this chapter. The approach followed in developing this model can be summarized in the following three steps:

1. Deriving an explicit function expressing the change in the snowpack in terms of the sum of net energy fluxes $\sum H_t$ and the sum of snowfall over the interval between two consecutive snowpack measurements, i.e:

$$d_m(t) - d_m(t - k) = F(\sum_{i=t-k+1}^{t} H_t(i), \sum_{i=t-k+1}^{t} p_s(i))$$  \hspace{1cm} (6.13)

where $k$ is the interval between the snowpack measurements in days. This step is presented in section 6.5.

2. The second step is to derive a function that expresses the sum of the net energy fluxes $\sum H_t$ and the sum of snowfall $\sum p_s$ in terms of the parameter to be updated. This step is presented in section 6.6 for the parameters $\eta_c$, $\Delta T_s$, $\eta_a$, and $T_r$. 

Fig. 6.6 Sensitivity of the snowpack to the snow formation parameter $T_r$. 

Chapter 6: A Snowpack Updating Model

111
3. Finally, the equation obtained from step(1) is combined with the one from step(2) to obtain an explicit function of the parameter in terms of two consecutive snowpack measurements. This step is also presented in section 6.6.

6.5 The Snowpack as a Function of the Net Energy Fluxes and the Snowfalls

In the UBC snowpack routine presented in section 6.2 the snowpack is calculated in a recursive method. This means that the snowpack can only be calculated from the previous snowpack, the net energy flux and the snowfall of the previous day. The recursive method is computationally convenient and efficient since there is no need to store previous information, and in addition its computation equations are less complex. However, it can not be used for updating the snowpack parameters from snowpack measurements if the measurement interval is longer than one day. So, accomplishing the first step presented in the previous section, a relationship has been developed in which the snowpack is expressed in terms of the snowfalls, the net heat fluxes, and the cold content between two consecutive snowpack measurements. This relationship can be presented as follows:

- Let $k$ be the measurement interval for two consecutive snowpack measurements $\bar{d}_m(t)$ and $\bar{d}_m(t_k)$, and let $\kappa_f(i), \kappa_p(i),$ and $\kappa_a(i)$ for $t - k < i < t$ be the snowpack thermal condition tags defined as follows:

  - For a frozen snowpack, i.e., $H_e(i) \leq 0.0$:
    \[
    \kappa_f(i) = 1, \kappa_p(i) = \kappa_a(i) = 0.0
    \]

  - For a melting snowpack, $H_e(i) > 0.0$:
    \[
    \begin{align*}
    \kappa_p(i) &= 1.0 \\
    \kappa_a(i) &= 0.0 \\
    \end{align*}
    \]
    \begin{align*}
    \text{For } H_e(i) < d_m(i - 1) + p_s(i)
    \end{align*}
\[
\begin{align*}
\kappa_p(i) &= 0.0 \\
\kappa_a(i) &= 1.0 \\
\end{align*}
\]

For \( H_e(i) > d_m(i - 1) + p_s(i) \)

- Let \( \kappa_{fp}(i) = \kappa_f(i) + \kappa_p(i) \), so \( \kappa_{fp}(i) = 1.0 \) means that the snowpack is either frozen or melting.

- Let \( \theta_1(i), \theta_2(i), \theta_3(i), \theta_4(i), \theta_5(i), \theta_6(i), \theta_7(t) \) and \( \theta_8(t) \) be defined as follows:

\[
\theta_1(i) = \begin{cases} 
\kappa_{fp}(i) & \text{for } i = t \\
\kappa_{fp}(i)\theta_1(i + 1) & \text{for } t - k + 1 \leq i \leq t - 1 
\end{cases}
\]

\[
\theta_2(i) = \begin{cases} 
\kappa_p(i) & \text{for } i = t \\
\theta_1(i + 1)\kappa_p(i) & \text{for } t = k + 1 \leq i \leq t - 1 
\end{cases}
\]

\[
\theta_3(i) = \begin{cases} 
0.0 & \text{for } i = t \\
\lambda\kappa_f(i - 1)\kappa_p(i + 1) & \text{for } i = t - 1 \\
\lambda\kappa_f(i)\theta_3(i + 1) & \text{for } t - k + 1 \leq i \leq t - 2 
\end{cases}
\]

\[
\theta_4(i) = \begin{cases} 
0.0 & \text{for } i = t \& t - 1 \\
\lambda\theta_1(i + 2)\kappa_p(i + 1)\kappa_f(i) & \text{for } t - k + 1 \leq i \leq t - 2 
\end{cases}
\]

\[
\theta_5(i) = \begin{cases} 
0.0 & \text{for } i = t, t - 1 \& t - 2 \\
\lambda\kappa_f(i)[\theta_5(i + 1) + \\
\lambda\kappa_f(i + 1)\kappa_p(i + 2)\theta_1(i + 3)] & \text{for } t - k + 1 \leq i \leq t - 3 
\end{cases}
\]

\[
\theta_6(i) = \theta_2(i) + \theta_3(i) + \theta_4(i) + \theta_5(i) \text{ for } t - k + 1 \leq i \leq t
\]
\[
\begin{align*}
\theta_f(t) &= \lambda \kappa_f(t-k)\theta_0(t-k+1) \\
\theta_b(t) &= \theta_1(t-k+1)
\end{align*}
\]

- The snowpack at time \(i = t\) can be expressed as follows:

\[
d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \sum_{i=t-k+1}^{t} \theta_6(i)H_f(i) - \theta_7(t)H_{cc}(t-k) + \theta_8(t)d_m(t-k)
\]

(6.14)

The derivation of this relationship has been presented in detail in Appendix C.

6.6 The Snowpack Parameters As a Function of the Snowpack

In this section the snowpack parameters \(\eta_c, \Delta T_s, \eta_a\) and \(T_r\) are expressed individually as functions of the snowpack.

6.6.1 The Cloud Cover Correction Factor \(\eta_c\)

Substituting equations 6.2 and 6.4 into equation 6.1 yields:

\[
H_f(t) = I_s(t)[1 - C(t)][1 - A(t)] + [-20 + 0.94T_a(t)][1 - C(t)] + 1.24T_c(t)C(t) + H_n
\]

(6.15)

where \(H_n = H_c + H_a + H_r\).

Rearranging the above equation to obtain \(H_f\) in terms of the cloud cover, \(C\), yields:

\[
H_f(t) = \alpha(t)C(t) + \beta(t)
\]

(6.16)

where \(\alpha(t) = 1.24T_c(t)(t) + 20 - 0.94T_a(t) - I_s(t)[1 - A(t)];\) and

\[
\beta(t) = I_s(t)[1 - A(t)] - 20 + 0.94T_a(t) + H_n
\]

Modifying the cloud cover by the correction factor \(\eta_c\) results in:

\[
H_f(t) = \eta_c\alpha(t)C(t) + \beta(t)
\]

(6.17)
Chapter 6. A Snowpack Updating Model

Substituting equation 6.17 in equation 6.14 yields:

\[
d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \sum_{i=t-k+1}^{t} \theta_6(i)\alpha(i)C(i)]\eta_c \\
- \sum_{i=t-k+1}^{t} \theta_6(i)\beta(i) - \theta_7(t)H_{cc}(t-k) + \theta_8(t)d_m(t-k)
\]

Rearranging the above equation yields:

\[
\eta_c = \frac{1}{\sum_{i=t-k+1}^{t} \theta_6(i)\alpha(i)C(i)}[\theta_6(t)d_m(t-k) - d_m(t) \\
- \theta_7(t)H_{cc}(t-k) + \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) \\
- \sum_{i=t-k+1}^{t} \theta_6(i)\beta(i)]
\]

(6.18)

Replacing the snowpacks \(d_m(t)\) and \(d_m(t-k)\) in Eq. 6.18 with the correspondent measured snowpacks \(\tilde{d}_m(t)\) and \(\tilde{d}_m(t_k)\) yields the \(\eta_c\) value that if was used in calculating the snowpack from time \(t-k\) would have produced calculated snowpack values equal to the measured. In other words, Eq. 6.18 can be used to calculate the value of the parameter \(\eta_c\) that corrects the calculated snowpacks to match the measured. The problem, however, is that the course of development of the snowpack thermal condition may change by changing the value of \(\eta_c\), resulting in different set of values of the tags \((\kappa_f, \kappa_p\) and \(\kappa_a)\) for different values of \(\eta_c\). This problem can be overcome by applying an iterative routine to determine the correct set of tags. Fortunately, the iterative procedure is usually quite short (maximum of three steps), and is not needed for many situations, for example if the snowpack is melting where \(\kappa_p = 1\) and \(\kappa_f = \kappa_a = 0\).

It should be emphasized that adjusting the value of \(\eta_c\) alone may not be sufficient to match the calculated snowpacks with the measured ones. This happens, for example, if the snowpack is frozen and the calculated snowpack is found to be less than the measured. In such situation the snowpack cannot be increased by changing the cloud cover.
6.6.2 The Cloud Cover Parameter $\Delta T_s$

Substituting equation 6.9 into equation 6.16 yields:

$$H_t(t) = \alpha(t)(1 - \frac{T_x(t) - T_n(t)}{\Delta T_s}) + \beta(t)$$

$$= -\alpha(t)(T_x(t) - T_n(t)) \frac{1}{\Delta T_s} + \alpha(t) + \beta(t)$$

$$= \alpha^*(t) \left( \frac{1}{\Delta T_s} \right) + \beta^*(t)$$

(6.19)

where

$$\alpha^*(t) = -\alpha(t)(T_x(t) - T_n(t)); \text{ and}$$

$$\beta^*(t) = \alpha(t) + \beta(t)$$

Substituting equation 6.19 into equation 6.14 yields:

$$d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \left[ \sum_{i=t-k+1}^{t} \theta_6(i)\alpha^*(i) \right] \frac{1}{\Delta T_s}$$

$$- \sum_{i=t-k+1}^{t} \theta_6(i)\beta^*(i) - \theta_7(t)H_{cc}(t - k) + \theta_8(t)d_m(t - k)$$

Rearranging the above equation yields:

$$\frac{1}{\Delta T_s} = \frac{1}{\sum_{i=t-k+1}^{t} \theta_6(i)\alpha^*(i)} \left[ \theta_8(t)d_m(t - k) - d_m(t) \right]$$

$$- \theta_7(t)H_{cc}(t - k) + \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i)$$

$$- \sum_{i=t-k+1}^{t} \theta_6(i)\beta^*(i)$$

(6.20)

The discussion on the application of Eq. 6.18 applies also to Eq. 6.20.

6.6.3 The Snow Albedo Correction Factor $\eta_a$

Rearranging equation 6.15

$$H_t(t) = \alpha^o(t)AL(t) + \beta^o(t)$$
where

\[\alpha^o(t) = -I_s(t)(1 - C(t)) \] and \n\[\beta^o(t) = (I_s(t) - 20 + 0.94T_a(t))(1 - C(t)) + 1.24T_c(t)C(t) + H_c + H_a + H_r\]

Modifying the snow albedo is by the correction factor \(\eta_a\) results in:

\[H_t(t) = \eta_a\alpha^o(t)A(t) + \beta^o(t) \tag{6.21}\]

Substituting equation 6.21 in equation 6.14 results in:

\[d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \sum_{i=t-k+1}^{t} \theta_6(i)\alpha^o(i)A(i)\eta_a - \sum_{i=t-k+1}^{t} \theta_6(i)\beta^o(i) - \theta_7(t)H_{cc}(t - k) + \theta_8(t)d_m(t - k)\]

Rearranging the above equation yields:

\[\eta_a = \frac{1}{\sum_{i=t-k+1}^{t} \theta_6(i)\alpha^o(i)A(i)} [\theta_6(t)d_m(t - k) - d_m(t) - \theta_7(t)H_{cc}(t - k) + \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \sum_{i=t-k+1}^{t} \theta_6(i)\beta^o(i)] \tag{6.22}\]

Eq. 6.22 can be used, similar to Eq. 6.18, to correct the calculated snowpacks to match the measured by adjusting the snow albedo correction factor, \(\eta_a\).

### 6.6.4 The Snow Formation Parameter \(T_r\)

Substituting equation 6.12 into equation 6.14 results in:

\[d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)[1 - \frac{T_2(i)}{T_r}]pp(i) - \sum_{i=t-k+1}^{t} \theta_6(i)H_t(i) - \theta_7(t)H_{cc}(t - k) + \theta_8(t)d_m(t - k)\]
Chapter 6. A Snowpack Updating Model

Rearranging the above equation results in:

\[
\frac{1}{T_r} = \frac{1}{\sum_{i=t-k+1}^{t} \theta_1(i) T_a(i) pp(i) } \left[ \theta_8(t) d_m(t - k) - d_m(t) \right. \\
- \left. \theta_7(t) H_{cc}(t - k) - \sum_{i=t-k+1}^{t} \theta_6(i) H_i(i) \right] + \sum_{i=t-k+1}^{t} \theta_1(i) pp(i) \]

(6.23)

Equation 6.23 has more limitations than equations 6.18, 6.20 and 6.22, since it can only be used when \( 0.0 < T_a < T_r \). This problem can be overcome by using an iterative method.

6.7 Algorithm of the Snowpack Updating Model

The procedure used in updating a snowpack parameter from on-line snowpack measurements is presented below:

1. During the course of calculating the snowpack, using the routine of the UBC watershed model (section 6.2), the snowpack thermal condition tags \( (\kappa_f, \kappa_p, \text{and } \kappa_a) \) are determined on a daily basis.

2. Once a new snowpack measurement becomes available (at time \( t \)), the variables \( \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7 \) and \( \theta_8 \) are back calculated, according to section 6.5, from time \( t \) to time \( t - k \), of the previous snowpack measurement.

3. One of the snowpack parameters, \( \eta_c, \Delta T_s, \eta_a, \text{or } T_r \) can then be updated using one of Eqs. 6.18, 6.20, 6.22 or 6.23, respectively.

4. The new value of the snowpack parameter is then used in recalculating the snowpack starting from time \( t - k \) and ending at time \( t \) and step 1 is repeated. If the new set of tags values matches the previous one, the next step is then executed; otherwise
steps 2, 3 and 4 are repeated.

5. The updated snowpack parameter is then used in the UBC watershed model to calculate the flow forecasts.

6. The previous steps are repeated whenever a new snowpack measurement becomes available.

6.8 Application of the Snowpack Updating Model

The snowpack updating model was used to update the cloud cover correction factor, $\eta_c$, using on-line snowpack measurements from the Mount Fidelity data station, British Columbia. The updated $\eta_c$ was then used in the UBC watershed model to forecast flows from the Illecillewaet basin.

During the period Oct. 1, 1969 to Sep. 30, 1970 the snowpack at Mount Fidelity was measured six times (see Fig. 6.7). Only the last three measurements were useful for updating the parameter $\eta_c$, since the first three were collected during the accumulation stage when the snowmelt is insignificant and therefore the cloud cover has little effect on the snowpack. However, using this limited number of snowpack measurements in updating has dramatically improved the snowpack forecasts as can be seen from Fig. 6.7. The values of $\eta_c$ for this period are shown in Fig. 6.8, from which it can be concluded that the cloud cover in the off-line UBC watershed model was consistently underestimated.

Using the updated values of $\eta_c$ for calculating river flows from the Illecillewaet basin resulted in a general improvement of the flow forecasts as can be seen from Fig. 6.9, and the increase in the value of the coefficient of efficiency, $E$, from 0.42 for the un-updated flow forecasts$^3$ to 0.58 for the updated flow forecasts. However, using the updated values

$^3$These forecasts were calculated using a version of the UBC watershed model different from the one used in Chap. 5.
Fig. 6.7 The measured, un-updated and updated snowpacks for 1969-1970.
Fig. 6.8 The updated values of the cloud cover parameter $\eta_c$ for 1969-1970.
Fig. 6.9 Hydrographs of the observed, un-updated and updated flow forecasts and their residuals for 1970.
of the parameter $\eta_c$ resulted in significant underestimation of the high flows between end of May and the middle of June. This is attributed to the fact that increasing the cloud cover to accommodate for the underestimation of the snowpack in the Fidelity station, which is located in the upper part of the Illecillewaet basin, resulted in the underestimation of the snowmelt in the lower parts of the Illecillewaet basin, which are the main sources of the flow at the beginning of the melting season.

The snowpack updating model was also used to update the parameter $\eta_c$ for the period Oct. 1, 1974 to Sep. 30, 1976. The first four snowpack measurements were collected during the accumulation stage and therefore were not used in updating. Fig. 6.10 shows great improvement in the snowpack forecasts after using the updated cloud cover correction factors shown in Fig. 6.11.

Using the updated factors in calculating the flow forecasts for the periods April 15 to Sep. 30, 1975 and April 15 to Sep. 30, 1976 resulted in a significant improvement in the flow forecasts as can be seen from Figs. 6.12 and 6.13, respectively. Evaluating these flow forecasts using the coefficient of efficiency, $E$, showed improvements in the value of the coefficient $E$ from 0.41 to 0.79 for the year 1975 and 0.25 to 0.61 for the year 1976.

6.9 Combined Application of the Snowpack and the Flow updating Models

The flow updating model presented in Chap. 5 was applied to update on a daily basis the flow forecasts calculated from the updated cloud cover correction factors $\eta_c$. Figs. 6.14, 6.15 and 6.16 show the updated flow forecasts using the snowpack updating alone, the flow forecasts updated by both the snowpack and the flow updating models, and their corresponding residuals for the periods April 15 to Sep. 30 for the years 1970, 1975 and 1976, respectively. All three figures show great improvements in flow forecasts due to the additional application of the flow updating model. Evaluating these flow forecasts
Fig. 6.10 The measured, un-updated and updated snowpacks for 1974-1976.
Fig. 6.11 The updated values of the cloud cover parameter $\eta_c$ for 1974-1976.
Fig. 6.12 Hydrographs of the observed flows, un-updated and updated flow forecasts and their residuals for 1975.
Fig. 6.13 Hydrographs of the observed flows, un-updated and updated flow forecasts and their residuals for 1976.
Fig. 6.14 Hydrographs of the observed flows, updated(snowpack only) and updated(snowpack and flow) flow forecasts and their residuals for 1970.
Fig. 6.15 Hydrographs of the observed flows, updated (snowpack only) and updated (snowpack and flow) flow forecasts and their residuals for 1975.
Fig. 6.16 Hydrographs of the observed flows, updated (snowpack only) and updated (snowpack and flow) flow forecasts and their residuals for 1976.
using the coefficient of efficiency shows improvements in the values of $E$ from 0.58 to 0.92 for the year 1970, from 0.79 to 0.93 for the year 1975, and from 0.61 to 0.88 for the year 1976.

Updating the cloud cover parameter $\eta_c$ may result in significant changes in the flow forecasts, that may lead to dramatic changes in the weighting factors of the flow updating model. Therefore, it is necessary to update the weighting factors of the flow updating model whenever the cloud cover parameter $\eta_c$ is updated.

6.10 Updating Parameters of the Snowpack Using Flow Measurements

The scarcity, if the not the absence, of snowpack measurements in many watersheds is a major problem for the application of the above snowpack updating model. For this reason, the theoretical structure of a model for updating snowpack parameters using flow measurements was laid out. In this model, on-line flow measurements are processed through a state-space model to update single parameters of the snowpack routine using the Kalman filter technique. The basic structure of this updating model and the derivation method are presented in Appendix D.

6.11 Summary and Conclusion

In this chapter, a study on the behavior of the snowpack, and the development of a method for updating the parameters of an energy budget snowpack model from on-line snowpack measurements are presented.

The snowpack was observed to pass through two distinctive stages: the accumulation and the depletion stages.

The snowpack accumulation stage is characterized by: intense snowfalls that result in high snow albedos and therefore reduce the shortwave energy inputs to the snowpack,
and very low temperatures that result in longwave, convective and advective heat losses from the snowpack. The net outcome is a freezing snowpack and a consistent increase in the snowpack.

During the snowpack depletion stage, the average temperatures are relatively high leading to longwave, convective and advective heat transfers to the snowpack, and the snowfalls are less in frequency and intensity, resulting in less snow albedo and therefore more shortwave heat transfer to the snowpack. The combined effect of these conditions result in a melting, and therefore depleting, snowpack.

The cloud cover affects both the shortwave and the longwave heat transfers to the snowpack; while the snow albedo affects the shortwave heat transfer only. Therefore, both the cloud cover and the snow albedo are functions of the snowmelt, which is induced by the net heat transfer to the snowpack. Since the snowmelt is significant only during the snowpack depletion stage, only the snowpack measurements collected during this stage can be used to update the estimates of the cloud cover and snow albedo. However, these snowpack measurements are important for updating the estimates of the snowfalls.

The effect of the cloud cover on the snowpack may not be distinguishable from that of the snow albedo, since both affects the shortwave heat transfer in a similar way. This may lead to difficulty in determining the source(s) of errors in the snowpack forecasts. Identifying the source(s) of the errors is important, since adjusting the value of a snowpack parameter to reduce a current snowpack forecast error may result in degrading the quality of the future snowpack forecasts.

An efficient method for identifying the sources of errors in the snowpack forecasts is to analyze the effect of synthesized errors in the snowpack parameters on the snowpack.

A model was developed to update the cloud cover estimates from snowpack measurements. The updating model was used on data from the Mount Fidelity station, and significant improvements were noticed for the three years under study.
The updated cloud cover estimates were then used in the UBC watershed model to calculate the flow forecasts for the Illecillewaet basin. Significant improvements in the flow forecasts of two years were achieved. In the other year, general improvement is noticed with the exception of the high flows at the beginning of the melting season. This was attributed to the overestimation of the cloud covers and therefore the underestimation of the snowmelt in the lower parts of the Illecillewaet basin which are the main sources of the flow at the beginning of the melting season.

Both the cloud cover updating model and the flow updating model presented in Chap. 5 were applied simultaneously on the flow forecasts, resulting in significant improvements in the flow forecasts for the three years of study.

Since updating the cloud cover estimates results in changes in the flow forecasts and therefore changes in the weighting factors of the flow updating model, it is necessary to update the weighting factors immediately after updating the cloud cover.
Chapter 7

Discussion and Conclusions

The main goal of this research is to develop techniques for supplying conceptual watershed models with feedback of real-time (on-line) hydrological data (e.g. flow and snowpack measurements) to update flow forecasts.

Since the main objective of any updating procedure is to reduce the residuals, or the errors, of the flow forecasts, the nature of these errors have been studied and techniques for analyzing these errors have been reviewed and discussed. In particular, a least squares criterion has been developed to test the linearity of forecast errors.

A review of many of the real-time flow forecasting models showed that most of the updating procedures used in these models are based on the Kalman filter technique. Theoretical and practical aspects of this technique have been investigated by using the method to update simulated flow forecasts, which were generated from a watershed model.

This review of the Kalman filter method showed that this technique is efficient in reducing linear errors. This important characteristic was exploited in developing a method for updating the flow forecasts of the UBC watershed model, which was then applied successfully to data from the Illecillewaet basin in British Columbia. The success of this method is based on the validity of the assumption of error linearity and the strength of the underlying UBC watershed model.

The above updating model aims at reducing the forecast errors after they were generated, i.e. it does not affect the source of these errors. Since snowmelt constitutes the
major input to the flows in the Illecillewaet basin, and mountainous watersheds in general, errors in estimating the snowpack may result in large flow forecast errors that may last for a long period of time, due to the long residence time of the snow. For this reason, a method was developed to update parameters of the UBC snowpack routine using real-time snowpack measurements, and then using these updated parameters in calculating flow forecasts. The updating method was applied successfully to meteorological data from Mount Fidelity station located in a high elevation of the Illecillewaet basin.

Both of the above mentioned updating methods were used in combination on data from the Illecillewaet basin. The results showed additional improvement in the flow forecasts. However, careful consideration should be given to the interaction between the two updating methods when these methods are used in combination.

The following is a discussion of the main findings of the thesis and the conclusions on the proposed methods, including a discussion of the combined use of these updating procedures.

7.1 Sources and Analysis of Forecast Residuals

The nature of the forecast errors, or the residuals, must be identified before a decision can be made on the type of updating procedure to be used. These forecast residuals can only be determined from historical data sets when the forecast flows are observed. Residuals of the flow forecasts are caused by one or a combination of the following: misrepresentative model structure, miscalibrated model parameters, errors in the input data; and errors in the reference data. Residuals caused by using a misrepresentative model structure are usually large and are best treated by redesigning or replacing the model. Forecast residuals due to miscalibration are better treated by recalibrating the model rather than designing a flow forecast updating procedure. However, forecast residuals
caused by errors in the model variables (e.g. the water storages, the snow albedo) may be effectively reduced using a model that updates the values of these variables. The effect of linear errors in the input data or the reference data can be effectively reduced using flow forecast updating models. On the other hand, misrepresentative data may produce forecast residuals which are difficult to correct, or to analyze.

Graphical presentation is an important means of evaluating the quality of the flow forecasts. Plots of the observed and the forecast hydrographs, and the residuals show the general performance of the forecasting model, and the distribution of the residuals and may reveal certain characteristics of the forecasts such as timing errors, or extreme errors. Errors in the flow rating curves may be evaluated through analyzing plots of the observed flow vs. the forecast, as has been shown in section 2.2.1. Correlation between consecutive residuals can be detected and evaluated through examining plots of the forecast residuals against previous residuals (see section 2.2.1,) and, more accurately, by examining plots of the autocorrelation and the partial autocorrelation functions, as has been displayed in section 2.2.3.

The quality of the flow forecasts can be evaluated using several least squares criteria. The coefficient of efficiency, $E$, measures the overall accuracy of the flow forecasts; while the coefficient of determination, $D$, measures the overall accuracy of the flow forecasts after removing linear errors. Based on these two coefficients, a least squares criterion $PLE$, defined in section 2.2.2, has been developed for evaluating linear errors in the flow forecasts. This combined use of the coefficients $E$ and $D$ appears to be a new criterion, and, although simple, is a very powerful tool. A high value of $PLE$ indicates that the flow forecasts are highly linear, and vice versa. The linearity of errors is an important condition for the application of the flow updating model presented in Chap. 5, and is also true for many of the updating techniques that can be found in the literature.

For water resource applications that require accurate, very short term, flow forecasts,
it is important to test the performance of the flow forecasting model against those of naive models. This is so, since the naive models are quite powerful on very short term forecasting and their applications require very minimal calculations. Two naive models were reviewed in this research. First the persistence model in which the flow is estimated equal to the most recent measured flow, and the extrapolation model in which the flow is estimated from the linear extrapolation of the last two measured flows. The performance of a flow forecasting model is tested against the performance of these two naive models using two least squares criteria: the coefficient of persistence $C_p$, and the coefficient of extrapolation $C_e$, respectively. Higher values of these criteria indicates that the flow forecasting model is doing better than the naive models, and vice versa. Both criteria have been discussed in Chap. 2.

It is important to note that since the persistence and the extrapolation models perform badly for long forecast leads, $C_p$ and $C_e$ are not suitable for evaluating long-term flow forecasts.

The discussion in section 2.2.2 regarding the characteristics of the coefficient of efficiency, the coefficient of persistence and the coefficient of extrapolation showed that the coefficient of efficiency is most suitable for evaluating flow forecasts.

The above least squares criteria can only be applied to historical data, which is not the basis for updating. However, these criteria indicate the nature of the errors, and can be used to evaluate the quality of the previous flow forecasts, and is hopefully a measure of future forecast accuracy.

7.2 Literature Review of the Real-Time Forecasting Models

Reviewing many of the existing real-time forecasting models showed that they can be classified into three types:
Chapter 7. Discussion and Conclusions

1. Error prediction models;

2. Transfer function models; and

3. Real-time conceptual models.

The error prediction models are not flow forecasting models *per se*, since they are basically procedures for estimating and correcting the errors in the flow forecasts of given flow forecasting models. In error prediction methods, the forecast residual is assumed to be related to its past values according to a certain time-series relationship (e.g. autoregressive process of order one $AR(1)$), as has been presented in section 3.1. This relationship is then used to estimate future values of errors, which are then used to adjust the flow forecasts. The performance of an error prediction method is very dependent on the correlations among consecutive errors. For this reason, this method is quite effective on the falling limb of the hydrograph, where correlations among consecutive errors are high; while being unpredictable on the rising limb where error correlations are weak. The performance of the error prediction method is also dependent on the forecast lead. Errors of short term flow forecasts are highly correlated, and therefore are more accurately estimated by the error prediction method; while correlations among errors of long term flow forecasts are weakly correlated, leading therefore to a weak performance of the error prediction method.

In the transfer function models, presented in section 3.3, the flow is estimated as a linear function of the input data as well as of some of the previously measured flows. The coefficients of this function are either fixed values that are previously determined by calibration using historical data, or variables that can be recursively estimated using an optimization technique (e.g. the Kalman filter, or the maximum likelihood technique). These models are sometimes referred to as black box or input-output models, since they simulate the relationship between the meteorological data as an input and the flow data as
an output with little consideration of the underlying physical processes. The performance of this type of model is dependent on the validity of the assumed linear relationship, which are generally based on the correlation among consecutive flows. These correlations are the strongest on the falling limb of the hydrograph and for short measurement intervals; while being weak on the rising limb and for long measurement intervals. The performance of a transfer function model is therefore expected to be superior on the falling limb and for short forecast leads, and unreliable on the rising limb and for long\textsuperscript{1} forecast leads. Two examples of this type of models are presented in section 3.3.

In the conceptual models, presented in section 3.4, the watershed system is visualized as a network of water storages, representing different soil layers, which receive input of rainfall and/or snowmelt, less losses, according to a priority system. In this system, the water storages representing the lower soil zones and the impervious areas have the highest priorities for the input, and the excess of the input is then transferred to the water storages representing the subsurface layer. The outflows from all water storages are assumed to be functions of the corresponding storages. These outflows can then be routed through a series of storages, and the sum of these routed flows is taken as the estimate of the flow at the basin outlet. The flow chart of a basic conceptual model is shown in Fig. 3.1. The parameters of a conceptual model can be treated as variables that can be updated from real-time data using the Kalman filter technique. The performance of a conceptual model is generally better than those of a transfer and an error prediction model, since it is based on a more realistic simulation of the watershed system. However, conceptual models are usually complex and more expensive to operate. Similar to the other types of real-time models, the conceptual models perform best on the falling limb.

\textsuperscript{1}Describing the forecast lead as short or long is based on the time of concentration of the watershed which is a function of the size and the topography of the watershed, the nature of the precipitation and the climate. In general, small, steep and rainfall-induced watersheds are characterized by short concentration times; while large, less steep and snowmelt-induced watersheds have long concentration times.
of the hydrograph and poorly on the rising limb. However, this behavior is caused by different reasons: the rising limb of the hydrograph is mainly controlled by the upper water storages which are highly variable and therefore difficult to predict; while the falling limb is driven by the lower water storages which change slowly and therefore are easier to estimate. Two examples of this type of model are shown in section 3.4.

The main characteristics of the above models can be summarized as follows:

- All the models perform well on the falling limb of the hydrograph.
- The performances of all models deteriorate rapidly on the rising limb.
- The error prediction models are very dependent on the correlation among consecutive flow forecast residuals; while transfer function models are dependent on the correlation among consecutive flow forecasts.
- The transfer function models do well only for short forecast leads.
- The conceptual models tend to out-perform the other two type of models as forecast lead increases.
- The timing error is not taken into consideration in any of the above models. These errors may lead the real-time model to degrade, rather than improve the flow forecasts.

7.3 The Kalman Filter

If the error analysis discussed in Chap. 2 and summarized in section 7.1 shows that there are significant linear errors, then two approaches are possible for improving flow forecasts: a) recalibration, and b) linear updating of the flow forecasts, which can be done
effectively using the Kalman filter technique, which is widely used by many real-time flow forecasting models.

The Kalman filter is a recursive\textsuperscript{2} technique that updates the estimate of an unknown quantity, usually referred to as a state, using real-time measurements. The state estimate is determined such that it produces the least sum of squares of errors and is unbiased\textsuperscript{3}, which are conditions necessary to obtain an optimal estimate based on the assumption that the state is a stochastic normally distributed variable. From this point of view, the Kalman filter can be seen as a Bayesian approach in which the estimates of the mean and the variance of the normal probability density function of the state are updated whenever a relevant measurement is available, with the mean being the optimal estimate of the state and the variance determining the confidence limits of this optimal estimate.

The updating procedure in the Kalman filter is based on calculating the updated estimate as a weighted sum of two estimates: one is the estimate calculated as a linear function, usually referred to as the state equation, of the previous values; and the other is the estimate calculated as a linear function, usually referred to as the measurement equation, of the measurement. The weighting factors are determined such that the conditions of the least sum of squares of errors and the unbiasedness of the estimates, discussed above, are satisfied. These weighting factors are functions of two important quantities in the Kalman filter theory, namely: the state noise variance and the measurement variance. These variances determine the accuracies of the estimates obtained from the state and the measurement equations, respectively, such that smaller variances represent higher quality estimates, and vice versa. The updating process in the Kalman filter is therefore dependent on two factors: the accuracy of estimating the state from its previous estimate, and the quality of the measurement, such that the better is the quality of the

\textsuperscript{2}A recursive technique is one that does not need to store past measurements.

\textsuperscript{3}Unbiased estimate is one that converges to the true value, if large number of measurements are available.
measurement, the more is the updating in the state estimate, and vice versa.

In this study, it has been proven, as has been presented in Chap. 4 that the updating process in the Kalman filter is a function of the relative, rather than the absolute, values of the above mentioned state noise and measurement variances\(^4\). Therefore, in applying sensitivity analysis for selecting the optimal values of these variances, it would be sufficient to test updating model performance against one variance only, while keeping the other fixed.

Based on a simulation study, presented in Chap. 4, in which linear errors are introduced in the flow forecasts, it was found that the state variability, measurement quality and the forecast lead are the main factors to be considered in selecting the optimal values of the state noise variance and the measurement variance. In general, large state noise and small measurement variances are recommended for a highly variable system, good measurement and a long forecast lead; while small state noise and large measurement variances are recommended for a slowly variable system, a poor measurement and a short forecast lead. However, very large state noise variances should be avoided since they may cause very large fluctuations in the state estimates as a result of poor measurements.

The adaptive Kalman filter approach is based on updating the state noise variance and the measurement variance using real-time data. Based on reviewing some of the methods using this approach, and the experience of the author from applying one of these methods, the adaptive Kalman filtering is not recommended. This is so for the following reasons:

1. Most of these adaptive methods are based on assuming that the state noise and the measurement variances are changing very slowly. Using these adaptive methods

\(^4\)This is true in the region beyond the influence of the initial conditions, or if the initial state variance, discussed in Chap. 4, is a function of the state noise variance.
would result therefore in insignificant changes in these variances, resulting therefore in minimal, if any, improvements in the flow forecasts.

2. Most of the adaptive models require strict conditions that can not be satisfied even for simple hydrological applications.

3. Larger number of measurements are needed as compared to the nonadaptive approach, which is a particularly serious problem for many hydrological applications.

4. Additional parameters are introduced, which need to be assigned, creating therefore additional sources of uncertainties in the forecasts;

5. Significant increase in the number of computations.

As stated above, a fundamental condition for applying the Kalman filter is linearity of the system. However, certain linearization techniques can be used to linearize nonlinear relationships, the Taylor expansion being a popular technique, but statistical linearization is a better and more consistent alternative, especially for highly nonlinear relationships (e.g. threshold-type ones).

7.4 The Proposed Updating Procedures

Based on analyzing the errors in the flow forecasts of the UBC watershed model and studying their main sources, two updating procedures were developed: a) in one, the flow forecasts of the UBC watershed model are directly updated from real-time flow measurements; and b) in the other, parameters of the snowpack routine in the UBC watershed model are updated using real-time snowpack measurements, and the updated parameters are then used in the UBC watershed model to upgrade the flow forecasts. In addition, both procedures can be used together in updating the flow forecasts. In the
following sections, these two updating procedures and their single and combined use are briefly presented and discussed.

7.4.1 Direct Updating of the Flow Forecasts

Analyzing the errors of the flow forecasts of the Illecillewaet basin generated by the UBC watershed model showed that they are mostly linear in structure. Since, as mentioned before, the Kalman filter is very effective in reducing linear errors, it was used in developing a procedure for updating the flow forecasts of the Illecillewaet basin. This updating procedure was based on the assumption that flow can be estimated as a weighted sum of the flow components calculated by the UBC watershed model, with the weighting factors being the states updated by the Kalman filter using real-time flow measurements. The main equations and the algorithm of this procedure are discussed in details in Chap. 5.

As mentioned before in section 7.3, the two most important parameters of the Kalman filter are the state noise variance and the measurement variance. For the current flow updating procedure, the state noise variance is a matrix, with its diagonal elements constituting the variances of the random noises associated with each weighting factor, and non-diagonal elements being the covariances between the noises of the weighting factors; while the measurement variance is simply the variance of a flow measurement\(^5\). The optimal values of these variances were selected from sensitivity analysis of the quality of the updated flow forecasts, evaluated in terms of the coefficient of efficiency \(E\), versus the measurement variance while fixing the state noise variance. Using one variance as a basis of sensitivity analysis, while fixing the other, was sufficient since, as mentioned before, the updating procedure in the Kalman filter is mainly a function of the relative not the absolute values of the state noise variance and the measurement variance. Two

---

\(^5\) The state noise and the measurement variances of this updating procedure are presented in details in Chap. 5.
assumptions were tested: one assumption is to select the state noise variance and the measurement variance as a fixed quantity throughout the updating period; and the other assumption is to select the state noise variance as a square of a certain percentage of the corresponding state, and the measurement variance as a square of a certain percentage of the most recent flow measurement. In general, the percentage form was found to be more efficient in updating high flow forecasts. This supports the observations of many researchers, namely that higher measurement errors are associated with higher flows, and therefore, the measurement variance is better selected as a direct function of the flow measurement. The sensitivity analysis showed that the measurement variance is best selected to lie between the squares of 10% and 20% of the flow measurement, and that the state noise variance components gave the best results if selected equal to 10% to their corresponding weighting factors.

The updating procedure was applied on the flow forecasts of the Illecillewaet basin for the period April 15 to July 15, 1970, which is characterized by high flows due to the snowmelt. The flow forecasts were updated for forecast leads of 1, 5, 10, 20, and 30 days. Significant improvements in the flow forecasts, measured by the coefficient of efficiency $E$, were achieved even for forecast leads up to 20 days. The limited improvement in the flow forecasts for a forecast lead of 30 days is due to the lack of enough measurements for updating (4 flow measurements for the period of three months). The ability of the updating model to improve the flow forecasts even for a long forecast lead of 20 days is mainly attributed to two factors: one is the validity of the assumption that the errors in the flow forecasts of the UBC watershed model are highly linear and of the form described by the updating procedure; and the other is the strength of the underlying UBC watershed model.

Using the adaptive method of Kitanidis and Bras (1978b) to update the state noise variance for the above updating procedure, resulted in no significant difference in the
flow forecasts. This is because this adaptive method was designed on the assumption that the state noise variance is slowly changing and therefore the state noise variance is not changed significantly enough to cause significant changes in the flow forecasts. The application of this adaptive method required significant increases in the complexity and the number of computations, and an additional effort for calibrating new parameters. For all these reason, this adaptive method is not recommend for the current, or similar, updating procedures. In fact, based on reviewing other forms of adaptive Kalman filtering, adaptive methods are generally not recommended for several reasons discussed before in section 7.3. However, the final decision on whether to use such methods for a particular application should be based on testing this application using several of these methods.

7.4.2 A Snowpack Updating Model

In the previous updating procedure, the flow forecasts were directly updated using real-time flow measurements, without affecting the source of errors in the watershed model. Since the snowmelt constitutes the main source of high flows for mountainous watershed systems, as in the Illecillewaet basin, it is expected that a high percentage of errors in the flow forecasts will be caused by errors in estimating the snowmelt. In addition, errors in the snowpack estimates are expected to result in long term errors in the flow forecasts due to the long residence time of the snow. For these reasons, an updating procedure was designed to update parameters of the snowpack routine of the UBC watershed model using real-time snowpack measurements, and then using the updated snowpack parameters in the UBC watershed model to calculate flow forecasts.

As a preliminary step in the design of the snowpack updating procedure, snowpack sensitivity to several snowpack parameters were analyzed. Testing two parameters associated with the estimation of the cloud cover showed that underestimating the cloud
Chapter 7. Discussion and Conclusions

cover results in insignificantly increasing the thickness of the snowpack during its accumulation stage and significantly increasing the rate of the snowmelt during the melting season. This is an important result, since mis-estimating the rate of the snowmelt results in timing errors and generally poor forecasts of the high flows, and the remaining unmelted snowpack can be subject to accumulation errors. Studying the effect of the snow albedo on the snowpack showed that the underestimation of the snow albedo results in the overestimation of the snowmelt rate, and vice versa. The similarity between the effects of the cloud cover and the snow albedo on the snowpack is attributed to the practically similar, yet physically different, roles of each factors in controlling the amount of shortwave energy absorbed by the snowpack which contributes to the snowmelt. The percentage of cloud cover determines the percentage of the shortwave radiation which reaches the snowpack; while the snow albedo determines the percentage of the shortwave radiation reflected by the snow. However, the cloud cover also controls the amount of longwave energy leaving the snowpack. The similarity between the effects of the cloud cover and the snow albedo may cause a problem in identifying the sources of errors in the snowpack estimates. This problem can be significant, since it may lead to an erroneous adjustment of a parameter as a result of an error caused by another parameter, which may result in degrading, rather than upgrading, future forecasts. The problem of identifying the source of errors in the snowpack forecasts may be solved by analyzing the effects of synthesized errors in individual snowpack parameters on the snowpack estimates, as have been presented in Chap. 6, and also by using measured data, such as the US Corps data.

The snowpack updating procedure was applied to update the cloud cover estimates using snowpack data from Mount Fidelity station located on the upper region of the Illecillewaet basin, and then using the updated cloud cover estimates in updating the snowpack estimates. Significant improvement in the snowpack estimates was achieved,
although very small number of snowpack measurements, 6 to 8 measurements a year, were available. In addition, only the snowpack measurements collected during the snowmelt season, which make only half of the measurements, were useful in updating the cloud cover. This is because the cloud cover only influences the snowmelt and not the snowfall, and therefore is only significant during the melting season. However, the snowpack measurements during the snowpack accumulation stage are important for updating the snowfall estimates. The updated cloud cover estimates showed that the cloud cover was probably being consistently underestimated in the UBC watershed model. This information can be useful for improving the estimation of the cloud cover in the UBC watershed model, and shows how the snowpack updating method can be used as a calibration tool of the watershed model.

Using the updated estimates of the cloud cover in the UBC watershed model to estimate the streamflows from the Illecillewaet basin for three years, resulted in a significant improvement in the flow forecasts for two of these years. However, the high flows at the beginning of the snow melting season of the third year were underestimated. This is attributed to the overestimation of the cloud cover and therefore the underestimation of the snowmelt in the lower parts of the Illecillewaet basin, which are the main sources of the flow at the beginning of the melting season and are not truly represented by the Mount Fidelity station located in the upper regions of the Illecillewaet basin. However, the flow forecasts for the rest of the snow melting season were improved, since they are driven by the snowmelt in the upper parts of the basin which are more closely represented by the Mount Fidelity station.

The current snowpack updating model is only the first step towards developing a more comprehensive updating model. The current updating model only updates individual parameters, i.e. it can not update more than one parameter from a single snowpack measurement. On-going research is aimed at defining a set of criteria that can be used
to define the percentage of updating in different parameters as a result of an error in the snowpack forecast. The updating model is based on the assumption that the measurement is perfectly correct, i.e. it does not take into account the measurement error. Since very few snowpack measurements are usually available for updating, the negative effect of bad measurements on the parameter estimates can be serious. In the current updating model, this problem can best be avoided by carefully studying the changes in the parameter due to new measurements and avoiding extreme adjustment of these parameters. However, additional research is needed to upgrade the present updating model and to take into consideration measurement errors.

7.5 Updating Snowpack Parameters Using Flow measurements

As has been mentioned above, the snowpack measurements in most watersheds are quite scarce, creating therefore a problem for applying the above snowpack updating model. For this reason, the main structure of a model for updating the snowpack parameters from real-time flow measurements has been laid out in Appendix D. This updating model also takes into account the errors in the measurement, since it is based on the Kalman filter technique. This updating model is currently under development.

7.6 Detection and Correction of Timing Errors

In all the reviewed real-time forecasting model, the timing errors were not taken into account. For this reason, a procedure was developed based on using lagged linear regression in estimating and correcting the timing errors in the flow forecasts, as discussed in section 2.2.2. Applying this procedure resulted in a little improvement in the flow forecasts. This limitation is attributed to the high variability of the timing errors. However, this updating procedure can be a useful tool for calibration.
7.7 Combined Use of Updating Techniques

The snowpack updating model discussed in Chap. 6 and the flow updating model discussed in Chap. 5 were used in conjunction to update the flow forecasts. The snowpack updating model was used to update the cloud cover estimates, which were then used to calculate the flow forecasts. A further updating of the flow forecasts was then made using the flow updating model. This resulted in additional improvement in the flow forecasts. Since adjusting the estimates of the cloud cover affects the weighting factors of the flow updating model, it is important to apply the flow updating model whenever a snowpack parameter is updated. This is not a significant problem, since flow measurements are more frequently available than the snowpack measurements, and therefore there is almost always a flow measurement available when a snowpack measurement is collected.

7.8 General

This research is based on the premise that real-time flow forecasting, or flow forecasting in general, should be based on a simulation of the physical processes which control the behavior of the watershed system, and such models are often referred to as conceptual. In contrast to the physically based models, the success of the transfer function models are solely dependent on the validity of the statistical relationships on which they are based. Since, these relationships are short lived, the transfer function models are not expected to perform well beyond very short forecast leads. It can even be argued that the naive models, like the persistence model, are better alternatives for very short forecast leads. Although conceptual models are generally better alternatives than the transfer function models due to their physically based structures, they are usually complex and require the calibration of a large set of parameters. To avoid this problem and since many watershed models are not designed to operate in real-time mode, this research was aimed
at developing techniques for updating the flow forecasts of such watershed models without introducing major changes into the structures of these models.

As a first step toward developing updating techniques, methods for analyzing flow forecast error signals were studied, and an apparently new criterion, PLE, was introduced to detect and evaluate linear errors in the flow forecasts. This is particularly important, since linearity of errors is a basic assumption in many updating techniques.

Then, the Kalman filter technique, used in many real-time forecasting models, was studied and presented using simulated data. The results from this study showed that the updating process in the Kalman filter is mainly a function of the relative, rather than the absolute, values of the state noise variance and the measurement variance. The optimal values of these variances are found to be functions of the variability of the parameters to be updated, the measurement quality and the forecast lead.

Because of its capability to reduce linear errors, the Kalman filter was the first approach used to develop a method for updating the flow forecasts of the UBC watershed model. This updating method was applied successfully on data from the Illecillewaet basin. The high performance of this updating method, even for long forecast leads, is mainly attributed to the high linearity of the flow forecast errors, and the strength of the underlying UBC watershed model.

Since high flows in the mountainous basins are mainly driven by the snowmelt, estimation of the snowpack establishes the volume of snowmelt runoff available during the season, and estimation of the melt rate determines the time distribution of runoff. Therefore, an updating procedure was developed to update parameters of the snowpack routine of the UBC watershed model using real-time snowpack measurements. This updating method was applied on data from the Mount Fidelity station, and then the updated parameters were used in the UBC watershed model to upgrade the snowpack and the flow forecasts. The snowpack forecasts were significantly improved and the flow
forecasts were generally improved, with the exception of one year where the flow was underestimated at the beginning of the melting season. This was attributed to the underestimation of the snowmelt in the lower regions of the Illecillewaet basin which were not truly represented by the Mount Fidelity station located in the upper region of the basin. Future upgrading of the snowpack updating procedure would include updating more than one parameter at a time, and taking into consideration errors in the snowpack measurements.

Both of the above updating techniques were used in combination, resulting in additional improvement in the flow forecasts. Since updating snowpack parameters affect the weighting factors in the flow updating model, the flow updating model should always be applied after any updating of the snowpack parameters.

To overcome the scarcity of the snowpack measurements, an updating procedure is suggested for updating parameters using real-time flow measurements. This updating procedure is at an early stage of development.

Since timing errors were virtually ignored in all reviewed updating procedures, a simple method for detecting and correcting timing errors based on lagged linear regression was suggested. This method has been applied with limited success on data from the Illecillewaet basin.
REFERENCES


REFERENCES


REFERENCES


REFERENCES

Appendix A

Hydrological and Climatic Characteristics of the Illecillewaet Watershed

The Illecillewaet river is a typical tributary to the upper Columbia river. The following paragraphs give a brief description of the hydrology and climate of this upper Columbia region.

The Columbia River flows to the northwest within a wide glaciated valley, known as the Rocky Mountain Trench, from its source in Columbia Lake for a distance of approximately 200 miles. It then makes an abrupt turn and flows in a southerly direction for about 100 miles to Revelstoke. The Canoe River, a major tributary, flows in a south-easterly direction within the Rocky Mountain Trench to meet the Columbia River just upstream of the site of Mica Dam. With the completion of Mica Dam and subsequent filling of the reservoir, McNaughton Lake, the Rocky Mountain Trench will be flooded from about Valemount to Donald, a distance of 130 miles.

The drainage boundary is formed by the Rocky Mountain chain in the east and by the Cariboo and Monashee Mountains in the west. The Purcell and Selkirk Mountains separate the northerly flowing section of the Columbia River from the section flowing to the south from Mica Dam. Several of the mountain peaks within the basin exceed El. 10,000 and rise more than 8000 feet above the valley floor. Some 10 percent of the drainage area, approximately 1000 square miles, is covered with permanent snowfields and glaciers.

The Columbia River basin between Revelstoke and Golden is densely forested with prominent stands of cedar and hemlock at the lower elevations and spruce and fir at the
higher elevations.

Vegetation thins above 5000 feet and the tree-line lies in the range of 6000 to 7000 feet. In the area just upstream of Golden, a medium to heavy Douglas fir and pine-grass vegetation cover is dominant. Further south, the forest cover changes to a lighter and more open mixture of Douglas fir, white spruce, yellow pine, larch, lodgepole pine and aspen in varying proportions.

The occurrence of several major lakes in the Columbia River system reflects the extent of major glaciation in the past. The only natural lakes of any significant size remaining upstream of Revelstoke are Columbia Lake and Windermere Lake in the headwater section of the Columbia River.

CLIMATE

The Columbia River basin, located some 300 to 400 miles inland from the Coast Mountains, is influenced primarily by the maritime air masses and by weather systems moving eastwards across the basin.

The temperature regime is strongly dependent on elevation, but temperatures recorded at several long-term meteorological stations located in the valleys indicate a fairly uniform pattern of temperature across the basin. Average monthly temperatures recorded at Revelstoke and several meteorological stations in the Rocky Mountain Trench are quite similar, ranging from lows of 10 to 15°F in January to highs of 60 to 65°F in July. The average diurnal temperature range at the valley stations displays a seasonal trend varying from about 15°F during the winter, to 30 to 35°F in July. Extreme maximum temperatures in excess of 100°F have been recorded at several stations within the basin.

Long-term meteorological records at Glacier (El. 4100) provide information on the temperatures in the middle elevation ranges in the basin and on the temperature lapse rates. The mean monthly temperatures in July are in the 55 to 60°F range, and in the winter are about 10°F.
The rates of change of temperature with elevation (lapse rates) based on a comparison of daily maximum and minimum temperatures, recorded at stations above elevations, show greater variability in the winter than in the summer. In the summer, the lapse rates in the maximum daily temperatures are consistently high, averaging about 4°F per 1000 feet; but, lapse rates in the minimum daily temperatures tend to be lower and more erratic because at night, the cooler air often drains into the valleys. During the winter months, lapse rates on both the daily maximum and minimum temperatures are low (averaging 1-2°F per 1000 feet), and variable, as arctic air frequently becomes trapped in the valleys with warmer maritime air circulating aloft.

Substantial snowpacks develop during the winter at all elevations in the basin. The snowpack in the valley bottom at Revelstoke is usually depleted by the end of April, but permanent snowfields exist at the highest elevations. The rapid snowpack depletion in the lower and middle elevation ranges during the late spring is the dominant influence on the hydrology of the region. The magnitude of the warming temperature pattern, but also on the distribution of the snowpack throughout the basin.

The eastward movement of moisture-laden maritime air masses across the Selkirk and Purcell Mountains produces considerably more precipitation in the drainage area downstream of Mica, than in the upstream area to the east of the mountains in the Rocky Mountain Trench.

Precipitation data indicates that the meteorological stations in the Columbia River valley downstream of Mica Dam not only receive significantly more precipitation than the stations in the Rocky Mountain Trench, but also a greater proportion of that precipitation falls as snow. The long-term average annual precipitation ranges from approximately 43 inches at Revelstoke (El. 1500) to 60 inches at Glacier (El. 4100) and to 16 inches at Canal Flats (El. 2650). The water equivalent of the average annual snowfall for the same stations are 16.2 inches, 38.2 inches and 4.4 inches respectively. Measurement of
Appendix A. The Illecillewaet Watershed

snowpacks above El. 6000 indicate that the water equivalent of the winter accumulation is in the order of 50 to 80 inches.
Appendix B

Detailed Algorithm of the Flow Updating Model

The algorithm of the flow updating model, presented in Chap. 5, is presented in details and sample calculations are provided.

B.1 Prediction

B.1.1 The Initial Conditions

Initially, the weighting factors and the covariance matrix are given the following values:

\[
\hat{X}(0)_- = \begin{bmatrix}
1.0 \\
1.0 \\
1.0
\end{bmatrix}; \text{ and }

\hat{P}(0)_- = \begin{bmatrix}
0.01 & 0.0 & 0.0 \\
0.0 & 0.01 & 0.0 \\
0.0 & 0.0 & 0.01
\end{bmatrix}
\]

These initial values are selected based on no previous knowledge of the weighting factors. The variances (diagonal elements) are simply taken as the square of 10% of the corresponding initial values of the weighting factors. Setting the covariances (the nondiagonal elements) equal to zero is based on the assumption that are no initial correlations between different weighting factors. Although this assumption is not true, its influence on the model performance is insignificant\(^1\) since the model updates the covariance matrix

\(^1\)Based on sensitivity analysis of the initial conditions.
Appendix B. Detailed Algorithm of the Flow Updating Model

as new flow measurements become available. The effect of the initial conditions can be dramatically reduced by running the updating model on a previous data record.

**B.1.2. For any time** \( t > 0.0 \)

The weighting factors and the covariance matrix are calculated from their previous values as follows:

\[
\begin{align*}
\hat{x}_{k+1}(-) &= \hat{x}_k(+) \\
\hat{P}_{k+1}(-) &= \hat{P}_k(+) + G_k
\end{align*}
\]

Eq. B.2 can be expanded as follows:

\[
\begin{align*}
\hat{P}_{k+1}(-) &= \begin{bmatrix}
\sigma_{ss}(t_k) + \sigma_{w_s}^2 & \sigma_{sr}(t_k) & \sigma_{sg}(t_k) \\
\sigma_{sr}(t_k) & \sigma_{rr}(t_k) + \sigma_{w_r}^2 & \sigma_{rg}(t) \\
\sigma_{sg}(t_k) & \sigma_{rg}(t) & \sigma_{gg}(t_k) + \sigma_{w_g}^2
\end{bmatrix}
\end{align*}
\]

The flow for the time \( t_{k+1} \) is then estimated as follows:

\[
Q_e(t_{k+1}) = \alpha_s(t_{k+1}) Q_s(t_{k+1}) + \alpha_r(t_{k+1}) Q_r(t_{k+1}) + \alpha_g(t_{k+1}) Q_g(t_{k+1})
\]

**B.2 Updating**

Once a new flow measurement becomes available the weighting factors and the covariance matrix are updated according to the following:

**B.2.1 The Kalman Gain**

The Kalman gain \( K(t_{k+1}) \) is calculated as follows:

\[
K_{k+1} = \hat{P}_{k+1}(-)H_{k+1}^T[H_{k+1}\hat{P}_{k+1}(-)H_{k+1}^T + R]^{-1}
\]
Appendix B. Detailed Algorithm of the Flow Updating Model

The term $\hat{P}_{k+1}(-)H_{k+1}^T$ can be expanded as follows:

$$
\hat{P}_{k+1}(-)H_{k+1}^T = 
\begin{bmatrix}
Q_s\sigma^2_{s-} + Q_s\sigma^2_{sr-} + Q_r\sigma^2_{sg-} \\
Q_s\sigma^2_{sr-} + Q_s\sigma^2_{xr-} + Q_r\sigma^2_{rg-} \\
Q_s\sigma^2_{sg-} + Q_s\sigma^2_{rg-} + Q_r\sigma^2_{sg-}
\end{bmatrix}
$$

(B.5)

The denominator in Eq. B.4 can be expressed as follows:

$$
D = Q_s^2(t_{k+1})\alpha_{ss}(t_{k+1}) - + Q_r^2(t_{k+1})\alpha_{sr}(t_{k+1}) - + Q_g^2(t_{k+1})\alpha_{sg}(t_{k+1}) - + \\
2[Q_s(t_{k+1})Q_r(t_{k+1})\alpha_{sr}(t_{k+1}) - + Q_s(t_{k+1})Q_g(t_{k+1})\alpha_{sg}(t_{k+1}) - + \\
Q_r(t_{k+1})Q_g(t_{k+1})\alpha_{rg}(t_{k+1}) -] + \sigma_v^2
$$

(B.6)

Substituting Eqs. B.5 and B.6 into Eq. B.4 yields:

$$
K_{k+1} = \frac{1}{D} 
\begin{bmatrix}
Q_s\alpha_{ss-} + Q_r\alpha_{sr-} + Q_g\alpha_{sg-} \\
Q_s\alpha_{sr-} + Q_r\alpha_{sr-} + Q_g\alpha_{rg-} \\
Q_s\alpha_{sg-} + Q_r\alpha_{rg-} + Q_g\alpha_{sg-}
\end{bmatrix}
$$

(B.7)

From the above equation it can be seen that Kalman gain is a function not only of the variances of each weighting factors but also of the covariances. This is an important fact that will be discussed in the following sections.

B.2.2 The Weighting Factors

The weighting factors are updated as follows:

$$
\hat{x}_{k+1}(+) = \hat{x}_{k+1}(-) - K_{k+1}r_{k+1}
$$

(B.8)

From Eq. B.8, it can be seen that the adjustment in a given weighting factor due to a new measurement is a function of its corresponding variance and covariances, the
Appendix B. Detailed Algorithm of the Flow Updating Model

The degree of this adjustment increases as the corresponding variance and flow component, and the forecast residual increase, and vice versa. On the other hand, the effect of the other flow components is dependent on the corresponding covariances. If this given weighting factor is negatively correlated (negative covariance) with another weighting factor, then the higher the flow component of the latter weighting factor is, the less the updating is in the former weighting factor, and vice versa. This explains why in some cases the response of a certain weighting factor to the forecast residual is much less or even opposite to what is expected. For example, an overestimation in the flow forecast is expected to result in an overall decrease of the weighting factors. However, as will be seen later in an example, high negative covariances between the weighting factors may lead to increases, rather than decreases, in the weighting factors of the less dominant flow components. The role of the covariances in the updating process is important to avoid over-adjusting the weighting factors.

B.2.3 The Covariance Matrix

The covariance matrix, \( \mathbf{P} \) is updated as follows:

\[
\hat{\mathbf{P}}_{k+1}(+) = [\mathbf{I} - \mathbf{S}_{k+1}] \hat{\mathbf{P}}_{k+1}(-)
\]  \hspace{1cm} (B.9)

where \( \mathbf{S} \) is a non-dimensional gain that can expressed as follows:

\[
\mathbf{S}_{k+1} = \mathbf{K}_{k+1} \mathbf{H}_{k+1}
\]  \hspace{1cm} (B.10)

Substituting Eq. B.7 into Eq. B.10 yields the following equation:

\[
\mathbf{S}_{k+1} = \begin{bmatrix}
K_1 Q_s & K_1 Q_r & K_1 Q_g \\
K_2 Q_s & K_2 Q_r & K_2 Q_g \\
K_3 Q_s & K_3 Q_r & K_3 Q_g
\end{bmatrix}
\]  \hspace{1cm} (B.11)
Substituting Eq. B.11 into Eq. B.9 results in:

\[
\hat{\mathbf{P}}_{k+1}(+) = \begin{bmatrix}
1 - K_1 Q_s & -K_1 Q_r & -K_1 Q_g \\
-K_2 Q_s & 1 - K_2 Q_r & -K_2 Q_g \\
-K_3 Q_s & -K_3 Q_r & 1 - K_3 Q_g
\end{bmatrix} \begin{bmatrix}
\alpha_{zz} & \alpha_{sr} & \alpha_{sg} \\
\alpha_{sr} & \alpha_{rr} & \alpha_{rg} \\
\alpha_{sg} & \alpha_{rg} & \alpha_{gg}
\end{bmatrix}
\]

(B.12)

where \( K_1, K_2 \) and \( K_3 \) are the elements of the Kalman gain matrix \( \mathbf{K} \). Eq. B.12 shows that the variances and covariances of a given weighting factor are updated, in a similar fashion to the weighting factors, as a function of the residual, their previous variance and covariances, and the flow components.

B.3 A Sample Calculation

The following example presents the procedures for estimating the flow on May 28, 1970 and then updating the weighting factors and the covariance matrix once the flow measurement becomes available.

B.3.1 Prediction

For calculating an estimate of the flow at May 28, 1970 the following weighting factors (calculated from previous day) are used:

\[
\mathbf{x}(\text{May } 28)_- = \begin{bmatrix}
\alpha_s \\
\alpha_r \\
\alpha_g
\end{bmatrix} = \begin{bmatrix}
0.457 \\
0.843 \\
1.199
\end{bmatrix}
\]

The corresponding covariance matrix is:

\[
\hat{\mathbf{P}}(\text{May } 28)_- = \begin{bmatrix}
0.1164 & -0.0490 & -0.0827 \\
-0.0490 & 0.3830 & -0.0149 \\
-0.0827 & -0.0149 & 0.1007
\end{bmatrix}
\]
Appendix B. Detailed Algorithm of the Flow Updating Model

The flow components calculated by the UBC watershed model for May 28, 1970 are as follows:

\[ \mathbf{H}^T(\text{May 28}) = \begin{bmatrix} Q_s \\ Q_r \\ Q_g \end{bmatrix} = \begin{bmatrix} 39.57 \\ 9.91 \\ 71.32 \end{bmatrix} \]

The flow for May 28, 1970 is estimated as follows (see Fig. 5.9c):

\[ Q_e(\text{May 28}) = \mathbf{H}(\text{May 28})\hat{\mathbf{X}}(\text{May 28}) = (0.457)(39.57) + (0.843)(9.91) + (1.199)(71.32) = 111.9 \left( \frac{m^3}{s} \right) \]

B.3.2 Updating

- The Forecast Residual

On May 28 the flow is measured and found to be equal to 96.8 \( \left( \frac{m^3}{s} \right) \). So, the forecast residual is calculated as follows:

\[ r(\text{May 28}) = Q_e(\text{May 28}) - Q_o(\text{May 28}) = 111.0 - 96.8 = 15.1 \left( \frac{m^3}{s} \right) \] (B.14)

- The Kalman Gain

The denominator in Eq. B.7, \( D \), is calculated using Eq. B.6 as follows:

\[ D = (39.57)^2(0.1164) + (9.91)^2(0.3830) + (71.32)^2(0.1007) + 2[(39.57)(9.91)(-0.0490) + (39.57)(71.32)(-0.0827) + (9.91)(71.32)(-0.0149)] + 200 = 405.81 \left( \frac{m^3}{s} \right)^4 \]
Kalman gain matrix is calculated from Eq. B.7 as follows:

\[
K = \frac{1}{405.81} \begin{bmatrix}
(39.57)(0.1164) + (9.91)(-0.0490) + (71.32)(-0.0827) \\
(39.57)(-0.0490) + (9.91)(0.3830) + (71.32)(-0.0149) \\
(39.57)(-0.0827) + (9.91)(-0.0149) + (71.32)(0.1002)
\end{bmatrix}
\]

this yields:

\[
K = \begin{bmatrix}
-4.38E-3 \\
1.96E-3 \\
9.18E-3
\end{bmatrix}
\]

The above calculations show that the Kalman gain element corresponding to the snowmelt weighting factor is negative as a result of the combined effect of the relatively large negative covariance between the snowmelt and groundwater weighting factors and the dominance of the groundwater flow component. Since the Kalman gain elements for the rain and the groundwater weighting factors are positive, these factors will be updated, as illustrated below, in the opposite direction of the snowmelt weighting factor.

• Updating the Weighting Factors

The weighting factors are updated according to Eq. B.8 as follows:

\[
\begin{bmatrix}
\hat{\alpha}_{s+} \\
\hat{\alpha}_{r+} \\
\hat{\alpha}_{g+}
\end{bmatrix} = \begin{bmatrix}
0.457 \\
0.843 \\
1.199
\end{bmatrix} - \begin{bmatrix}
-4.38E-3 \\
1.96E-3 \\
9.18E-3
\end{bmatrix} = 15.1
\]

(B.15)

The above equation yields:

\[
\begin{bmatrix}
\hat{\alpha}_{s+} \\
\hat{\alpha}_{r+} \\
\hat{\alpha}_{g+}
\end{bmatrix} = \begin{bmatrix}
0.523 \\
0.813 \\
1.060
\end{bmatrix}
\]

(B.16)
So, overestimating the flow resulted in the reduction of the rain and the groundwater weighting factors. However, the snowmelt weighting factor was increased mainly due to the influence of the groundwater weighting factor. This was found to improve the forecast of the hydrograph peak (see Fig. 5.9d.)

- **Updating The Covariance Matrix**

The covariance matrix is updated according to Eq. B.12 as follows:

\[
S(\text{May 28}) = \begin{bmatrix}
-0.1733 & -0.0434 & -0.3124 \\
0.0776 & 0.0194 & 0.1398 \\
0.3633 & 0.0910 & 0.6547
\end{bmatrix}
\]

\[
\bar{P}(\text{May 28})_+ = \begin{bmatrix}
1.1733 & 0.0434 & 0.3124 \\
-0.0776 & 0.9806 & -0.1398 \\
-0.3633 & -0.0910 & 0.3453
\end{bmatrix}
\begin{bmatrix}
0.1164 & -0.0490 & -0.0827 \\
-0.0490 & 0.3830 & -0.0149 \\
-0.0827 & -0.0149 & 0.1007
\end{bmatrix}
\]

Multiplying the above matrices results in:

\[
\bar{P}(\text{May 28})_+ = \begin{bmatrix}
0.1086 & -0.0455 & -0.0663 \\
-0.0455 & 0.3815 & -0.0222 \\
-0.0663 & -0.0222 & 0.0662
\end{bmatrix}
\]

It can be seen that the variances of the weighting factors after the measurement has been available (diagonal elements of matrix \(P_+\)) are smaller than those before the measurement is available (diagonal elements of matrix \(P_-\)). This shows the increase in confidence in estimating the weighting factors as a result of the additional information obtained from the new measurement, which is a basic characteristic of the Bayesian approach\(^3\).

\(^3\)See section 4.2.
B.3.3 Prediction for the Following Day

- The weighting Factors

The weighting factors are assumed, according to Eq. B.1, equal to the updated values in the previous day. So, the weighting factors for May 29, 1970 are predicted as follows:

$$\hat{x}(\text{May 29})_+ = \hat{x}(\text{May 28})_+$$

So, for May 29:

$$\begin{bmatrix} \hat{\alpha}_{s-} \\ \hat{\alpha}_{r-} \\ \hat{\alpha}_{g-} \end{bmatrix} = \begin{bmatrix} 0.523 \\ 0.813 \\ 1.060 \end{bmatrix}$$

- The Covariance Matrix

The covariance matrix is updated according to Eq. B.2 as follows:

$$\hat{P}(\text{May 29})_- = \hat{P}(\text{May 28})_+ + G$$

Substituting the values of $\hat{P}(\text{May 28})$ and $G$ in the above equation yields:

$$\begin{bmatrix} 0.1086 & -0.0455 & -0.0663 \\ -0.0455 & 0.3815 & -0.0222 \\ -0.0663 & -0.0222 & 0.0662 \end{bmatrix} + \begin{bmatrix} 0.01 & 0.0 & 0.0 \\ 0.0 & 0.01 & 0.0 \\ 0.0 & 0.0 & 0.01 \end{bmatrix}$$

This results in:

$$\hat{P}(\text{May 29})_- = \begin{bmatrix} 0.1186 & -0.0455 & -0.0663 \\ -0.0455 & 0.3915 & -0.0222 \\ -0.0663 & -0.0222 & 0.0762 \end{bmatrix}$$

As can be seen from the above equation, the prediction step results in increasing the variances of the weighting factors, which is interpreted as an increase in the
uncertainty of the future estimates of the weighting factors. As a new flow measurement becomes available, the weighting factors are updated as shown previously and their corresponding variances are then decreased (Eq. B.12), indicating an increase in the confidence of the estimates as a result of acquiring the new information. So, the change in a given weighting factor variance is a balance between the gain during the updating stage and the loss during the prediction stage. This balance is controlled by the measurement and the state noise variances, $R$ and $G$, respectively. The higher the quality of the measurement, the less is the measurement variance and the more are the decreases in the variances of the weighting factors during the updating stage, and vice versa. During the prediction stage, the larger the state noise variances the larger are the increases in the variances of the weighting factors, and vice versa.
Appendix C

Derivation of The Snowpack Function

The detailed derivation of the snowpack function used in Chap. 6 is presented. This function expresses the snowpack in terms of the sum of the energy fluxes and the snowfalls for the period between snowpack measurements. The change in the observed snowpack between times \( t - k \) and \( t \) can be expressed as follows:

\[
\tilde{d}_m(t) - \tilde{d}_m(t - k) = F \left[ \sum_{i=t-k+1}^{t} H_t(i) + \sum_{i=t-k+1}^{t} p_s(i) \right]
\]

Let the snowpack condition tags \( \kappa_f, \kappa_p, \) and \( \kappa_a \) be as defined in section 6.5, then the snowpack energy content, the snowpack water equivalent and the snowmelt can be expressed as follows:

Snowpack Energy Content

\[
H_{cc}(t) = \kappa_f(t)H_e(t) + \kappa_p(t)0.0 + \kappa_a(t)0.0
\]

\[
= \kappa_f(t)H_e(t)
\]  

(C.1)

Snowpack

\[
d_m(t) = \kappa_f(t)[d_m(t - 1) + p_s(t)] + \kappa_p(t)[d_m(t - 1) + p_s(t)]
\]

\[
- H_e(t)] + \kappa_a(t)0.0
\]

This results in:

\[
d_m(t) = \kappa_f(p) p_s(t) - \kappa_p(t)H_e(t) + \kappa_f(t)d_m(t - 1)
\]  

(C.2)

173
Appendix C. Derivation of The Snowpack Function

where $\kappa_{fp}(t) = \kappa_f(t) + \kappa_p(t)$

The Snowmelt

$$m(t) = \kappa_f(t)0.0 + \kappa_p(t)H_e(t) + \kappa_a(t)[p_s(t) + d_m(t - 1)]$$

this can be written as follows:

$$m(t) = \kappa_p(t)H_e(t) + \kappa_a(t)[p_s(t) + d_m(t - 1)] \quad (C.3)$$

From Eq. C.2, for $k = 1$:

$$d_m(t) = \kappa_{fp}(t)p_s(t) - \kappa_p(t)[\lambda\kappa_f(t - 1)H_{cc}(t - 1) + H_t(t)]$$

$$+ \kappa_{fp}(t)d_m(t) \quad (C.4)$$

Replacing the argument $t$ with $t - 1$ in Eq. C.4 results in:

$$d_m(t - 1) = \kappa_{fp}(t)p_s(t - 1) - \kappa_p(t - 1)[\lambda\kappa_f(t - 2)H_{cc}(t - 2)$$

$$+ H_t(t - 1)] + \kappa_{fp}(t - 1)d_m(t - 2)$$

Substituting the above equation into Eq. C.4 yields:

For $k = 2$

$$d_m(t) = \kappa_{fp}(t)p_s(t) + \kappa_{fp}(t)\kappa_{fp}(t - 1)p_s(t - 1) - \kappa_p(t)H_t(t)$$

$$- \kappa_{fp}(t)\kappa_p(t - 1)H_t(t - 1) - \lambda\kappa_p(t)\kappa_f(t - 1)H_{cc}(t - 1)$$

$$- \lambda\kappa_{fp}(t)\kappa_p(t - 1)\kappa_f(t - 2)H_{cc}(t - 2)$$

$$+ \kappa_{fp}(t)\kappa_{fp}(t - 1)d_m(t - 2) \quad (C.5)$$

Eq. C.5 can be generalized for $k \geq 2$:

$$d_m(t) = \sum_{i=t-k+1}^{t} (\prod_{j=i}^{t} \kappa_{fp}(i))p_s(i) - \kappa_p(t)H_t(t)$$
Appendix C. Derivation of The Snowpack Function

\[ - \sum_{i=t-k+1}^{i=t-1} ( \prod_{j=i+1}^{t} \kappa_{fp}(j)) \kappa_{p}(i) H_{t}(i) \]
\[ - \lambda \kappa_{p}(t) \kappa_{f}(t-1) H_{cc}(t-1) \]
\[ - \sum_{i=i-k}^{t-2} \lambda ( \prod_{j=i+2}^{t} \kappa_{fp}(j)) \kappa_{p}(i+1) \kappa_{f}(i) H_{cc}(i) \]
\[ + ( \prod_{i=t-k+1}^{t} \kappa_{fp}(i)) d_{m}(t-k) \]  

(C.6)

From Eq. C.1:

\[ H_{cc}(t) = \kappa_{f}(t) H_{n}(t) \]
\[ = \kappa_{f}(t)[\lambda \kappa_{f}(t-1) H_{cc}(t-1) + H_{t}(t)] \]  

(C.7)

Replacing \( t \) with \( t - 1 \) in Eq. C.7 yields:

\[ H_{cc}(t - 1) = \kappa_{f}(t - 1)[\lambda \kappa_{f}(t - 2) H_{cc}(t - 2) + H_{t}(t - 1)] \]  

(C.8)

Substituting Eq. C.8 into Eq. C.7 results in:

\[ H_{cc}(t) = \lambda^{2} \kappa_{f}(t) \kappa_{f}(t - 1)^{2} \kappa_{f}(t - 2) H_{cc}(t - 2) + \kappa_{f}(t) H_{t}(t) \]
\[ + \lambda \kappa_{f}(t) \kappa_{f}(t - 1)^{2} H_{t}(t - 1) \]  

(C.9)

For \( t - k + 1 \leq i \leq t \):

\[ H_{cc}(i) = \lambda^{2} \kappa_{f}(i) \kappa_{f}(i - 1)^{2} \kappa_{f}(i - 2) H_{cc}(i - 2) + \kappa_{f}(i) H_{t}(i) \]
\[ + \lambda \kappa_{f}(i) \kappa_{f}(i - 1)^{2} H_{t}(i - 1) \]

\( H_{cc}(i) \) can be expressed in terms of the snowpack variables from time \( i \) down to time \( i - 3 \) as follows:

\[ H_{cc}(i) = \lambda^{3} \kappa_{f}(i) \kappa_{f}(i - 1)^{2} \kappa_{f}(i - 2)^{2} \kappa_{f}(i - 3) H_{cc}(i - 3) \]
\[ + \kappa_{f}(i) H_{t}(i) + \lambda \kappa_{f}(i) \kappa_{f}(i - 1)^{2} H_{t}(i - 1) \]
\[ + \lambda^{2} \kappa_{f}(i) \kappa_{f}(i - 1)^{2} \kappa_{f}(i - 2)^{2} H_{t}(i - 2) \]
In general, $H_{cc}(i)$ can be expressed in terms of the snowpack variables from time $i$ down to time $t - k$ as follows:

$$H_{cc}(i) = \lambda^{i-t+k} \kappa_f(i) \kappa_f(t - k) \left( \prod_{j=t-k+1}^{i-1} \kappa_f(j) \right) H_{cc}(t - k)$$

$$+ \kappa_f(i) [H_t(i) + \sum_{j=t-k+1}^{i-1} \lambda^{i-j} \left( \prod_{l=j}^{i-1} \kappa_f(l)^2 \right) H_t(j)]$$

Since $\kappa_f(i) = 0.0$ or $1.0$, $\kappa_f(i)^2 = \kappa_f(i)$. So, the above equation can be written as follows:

$$H_{cc}(i) = \lambda^{i-t+k} \left( \prod_{j=t-k}^{i} \kappa_f(j) \right) H_{cc}(t - k) + \kappa_f(i) [H_t(i)$$

$$+ \sum_{j=t-k+1}^{i-1} \lambda^{i-j} \left( \prod_{l=j}^{i-1} \kappa_f(l) \right) H_t(j)]$$

(Sub. Eq. C.10 into the fourth term of the right-hand side of Eq. C.6 (for $i = t - 1$) yields:

For $k > 2$

$$\lambda \kappa_p(t) \kappa_f(t - 1) H_{cc}(t - 1) = \lambda^k \kappa_p(t) \left( \prod_{i=t-k}^{t-1} \kappa_f(i) \right) H_{cc}(t - k) + \lambda \kappa_p(t) \kappa_f(t - 1) [H_t(t - 1)$$

$$+ \sum_{i=t-k+1}^{t-2} \lambda^{t-i-1} \left( \prod_{j=i}^{t-2} \kappa_f(j) \right) H_t(i)]$$

(Sub. Eq. C.10 into the fifth term of the right-hand side of Eq. C.6 yields:

For $k = 2$

$$\lambda \kappa_p(t) \kappa_f(t - 1) H_{cc}(t - 1) = \lambda^2 \kappa_p(t) \kappa_f(t - 1) \kappa_f(t - 2) H_{cc}(t - 2)$$

$$+ \lambda \kappa_p(t) \kappa_f(t - 1) H_t(t - 1)$$

(Sub. Eq. C.10 into the fifth term of the right-hand side of Eq. C.6 yields:
Appendix C. Derivation of The Snowpack Function

For $k > 3$

\[
\sum_{i=t-k}^{t-2} \lambda \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_p(i+1) \kappa_f(i) H_{cc}(i) = \\
+ [\lambda \left( \prod_{i=t-k+2}^{t} \kappa_{fp}(i) \right) \kappa_p(t-k+1) \kappa_f(t-k)] \\
+ \lambda^2 \left( \prod_{i=t-k+3}^{t} \kappa_{fp}(i) \right) \kappa_p(t-k+2) \kappa_f(t-k+1) \kappa_f(t-k) \\
+ \sum_{i=t-k+2}^{t-2} \lambda^{i-t+k+1} \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_p(i+1) \left( \prod_{j=t-k}^{i} \kappa_f(j) \right) H_{cc}(t-k) \\
+ \lambda \left( \prod_{i=t-k+3}^{t} \kappa_{fp}(i) \right) \kappa_p(t-k+2) \kappa_f(t-k+1) H_t(t-k+1) \\
+ \sum_{i=t-k+2}^{t-2} \left\{ [\lambda \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_p(i+1) \kappa_f(i+1)] H_t(i) \\
+ \sum_{j=t-k+1}^{i-1} \lambda^{i-j} \left( \prod_{l=j}^{i-1} \kappa_f(l) \right) H_t(j) \right\} \\
= (C.13)
\]

For $k = 3$

\[
\sum_{i=t-k}^{t-2} \lambda \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_p(i+1) \kappa_f(i) H_{cc}(i) = \\
[\lambda \left( \prod_{i=t-1}^{t} \kappa_{fp}(i) \right) \kappa_p(t-2) \kappa_f(t-3)] \\
+ \lambda^2 \kappa_{fp}(t) \kappa_p(t-1) \kappa_f(t-2) \kappa_f(t-3) H_{cc}(t-3) \\
+ \lambda \kappa_{fp}(t) \kappa_p(t-1) \kappa_f(t-2) H_t(t-2) \\
= (C.14)
\]

For $k = 2$

\[
\sum_{i=t-k}^{t-2} \lambda \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_p(i+1) \kappa_f(i) H_{cc}(i) = \lambda \kappa_{fp}(t) \kappa_p(t-1) \kappa_f(t-2) H_{cc}(t-2) \\
= (C.15)
\]
Appendix C. Derivation of The Snowpack Function

Substituting Eqs. C.12 and C.15 into Eq. C.6 yields for \( k = 2 \):

\[
d_m(t) = \sum_{i=t-1}^{t} \left( \prod_{j=i}^{t} \kappa_{fp}(j) \right) p_s(i) - \kappa_p(t) H_t(t) - \left[ \kappa_{fp}(t) \kappa_p(t - 1) \right. \\
+ \lambda \kappa_f(t - 1) \kappa_p(t) \left] H_t(t - 1) - \lambda \kappa_f(t - 2) \left[ \lambda \kappa_p(t) \kappa_f(t - 1) \right. \\
+ \kappa_{fp}(t) \kappa_p(t - 1) \] H_{cc}(t - 2) + \left( \prod_{i=t-1}^{t} \kappa_{fp}(i) \right) d_m(t - 2)
\]

(C.16)

Substituting Eqs. C.11 and C.14 into Eq. C.6 yields for \( k = 3 \):

\[
d_m(t) = \sum_{i=t-2}^{t} \left( \prod_{j=i}^{t} \kappa_{fp}(j) \right) p_s(i) - \kappa_p(t) H_t(t) - \left[ \lambda \kappa_f(t - 1) \kappa_p(t) \right. \\
+ \kappa_{fp}(t) \kappa_p(t - 1) \left] H_t(t - 1) - \lambda^{2} \left( \prod_{i=t-2}^{t-1} \kappa_{f}(i) \right) \kappa_p(t) \right. \\
+ \lambda \kappa_{f}(t) \kappa_{f}(t - 2) \kappa_p(t - 1) + \left( \prod_{i=t-1}^{t} \kappa_{fp}(i) \right) \kappa_p(t - 2) \left] H_t(t - 2) \right. \\
- \lambda \kappa_{f}(t - 3) \lambda^{2} \left( \prod_{i=t-2}^{t-1} \kappa_{f}(i) \right) \kappa_p(t) + \lambda \kappa_{f}(t) \kappa_{f}(t - 2) \kappa_p(t - 1) \\
+ \left( \prod_{i=t-1}^{t} \kappa_{fp}(i) \right) \kappa_p(t - 2) \right] H_{cc}(t - 3) \\
+ \left( \prod_{i=t-2}^{t} \kappa_{fp}(i) \right) d_{m}(t - 3)
\]

(C.17)

Substituting Eqs. C.11 and C.13 into Eq. C.6 and rearranging yields for \( k \geq 4 \):

\[
d_m(t) = \sum_{i=t-k+1}^{t} \left( \prod_{j=i}^{t} \kappa_{fp}(j) \right) p_s(i) - \kappa_p(t) H_t(t) - \left[ \kappa_{fp}(t) \kappa_p(t - 1) \right. \\
+ \lambda \kappa_p(t) \kappa_f(t - 1) \left] H_t(t - 1) - \left[ \prod_{i=t-1}^{t} \kappa_{fp}(i) \right] \kappa_p(t - 2) \right. \\
+ \lambda^{2} \kappa_p(t) \left( \prod_{i=t-2}^{t-1} \kappa_{f}(i) \right) + \lambda \kappa_{f}(t) \kappa_p(t - 1) \kappa_f(t - 2) \right] H_t(t - 2) \\
- \left[ \prod_{i=t-2}^{t} \kappa_{fp}(i) \right] \kappa_p(t - 3) + \lambda^{3} \kappa_p(t) \left( \prod_{i=t-3}^{t-1} \kappa_{f}(i) \right) \\
+ \lambda \kappa_{f}(t) \kappa_{f}(t - 1) \kappa_p(t - 2) \kappa_f(t - 2)
\]
\[ + \lambda^2 \kappa_{fp}(t)\kappa_{p}(t-1)(\prod_{i=t-3}^{t-2} \kappa_f(i))H_t(t-3) \]

\[ - \left( \prod_{j=i+1}^{t} \kappa_{fp}(j) \right) \kappa_{p}(i) + \lambda^{t-i} \kappa_{p}(t) \left( \prod_{j=i}^{t-1} \kappa_f(j) \right) \]

\[ + \lambda \left( \prod_{j=i+1}^{t} \kappa_{fp}(j) \right) \kappa_{p}(i+1) \kappa_f(i) \]

\[ + \sum_{j=1}^{t-2-i} \lambda^{t-j-i} \left( \prod_{l=j}^{t} \kappa_{fp}(l) \right) \kappa_{p}(t-j) \left( \prod_{l=j}^{t-j-1} \kappa_f(l) \right) H_t(i) \]

\[ - \left( \prod_{i=t-k+2}^{t} \kappa_{fp}(i) \right) \kappa_{p}(t-k+1) + \lambda^{k-1} \kappa_{p}(t) \left( \prod_{i=t-k+1}^{t-1} \kappa_f(i) \right) \]

\[ + \lambda \left( \prod_{i=t-k+3}^{t} \kappa_{fp}(i) \right) \kappa_{p}(t-k+2) \kappa_f(t-k+1) \]

\[ + \sum_{i=1}^{k-3} \lambda^{k-i-1} \left( \prod_{j=t-k+1}^{t} \kappa_{fp}(j) \right) \kappa_{p}(t-i) \left( \prod_{j=t-k+1}^{t-i-1} \kappa_f(j) \right) H_t(t-k+1) \]

\[ - \lambda \kappa_f(t-k) \left( \prod_{i=t-k+1}^{t} \kappa_{fp}(i) \right) \kappa_{p}(t-k+1) + \lambda^{k-1} \kappa_{p}(t) \left( \prod_{i=t-k+1}^{t-1} \kappa_f(i) \right) \]

\[ + \lambda \left( \prod_{i=t-k+3}^{t} \kappa_{fp}(i) \right) \kappa_{p}(t-k+2) \kappa_f(t-k+1) \]

\[ + \sum_{i=t-k+2}^{t-2} \lambda^{i-t+k} \left( \prod_{j=i+2}^{t} \kappa_{fp}(j) \right) \kappa_{p}(i+1) \left( \prod_{j=t-k+1}^{i} \kappa_f(j) \right) H_{cc}(t-k) \]

\[ + \left( \prod_{i=t-k+1}^{t} \kappa_{fp}(i) \right) d_m(t-k) \quad \text{(C.18)} \]

Let \( \theta_1(i), \theta_2(i), \theta_3(i), \theta_4(i), \theta_5(i), \theta_6(i), \theta_7(t) \) and \( \theta_8(t) \) be defined as in section 6.5. Eq. C.18 can be expressed as follows:

\[
d_m(t) = \sum_{i=t-k+1}^{t} \theta_1(i)p_s(i) - \sum_{i=t-k+1}^{t} \theta_6(i)H_t(i) - \theta_7(t)H_{cc}(t-k) + \theta_8(t)d_m(t-k) \quad \text{(C.19)}
\]
Appendix C. Derivation of The Snowpack Function

Eq. C.19 is the same as Eq. 6.14 in section 6.5.
Appendix D

Derivation of the Flow-Snowpack Updating Model

D.1 Development of the Basic Model

In situations where the snowmelt constitutes the major input to the streamflow and the rainfall-generated flow component is insignificant, or can be easily separated, real-time flow measurements can be utilized to update the parameters of the snowmelt function. The approach is based on the following convolution equation:

\[
Q_s(t) = \sum_{i=1}^{k} \gamma(t + 1 - i)m(t + 1 - i)q_s(i)
\]  \hspace{1cm} (D.1)

where \( Q_s \) is the snowmelt flow component;

\( q_s(1), q_s(2), ..., q_s(k) \) are the indices of the snowmelt unit hydrograph;

\( m(i) \) is the snowmelt at day \( i \); and

\( \gamma \) is the percentage of the snowmelt that is routed to the snowmelt flow component (the rest is routed to the groundwater.)

Selecting the percentage of cloud cover as the main variable to be updated, the snowmelt for a given day, \( t \), can be estimated\(^1\) as follows:

\[
m(t) = \alpha(t)C(t) + \beta(t), \quad \text{For a partially melting snowpack;}
\]

\[
= d_m, \quad \text{For a completely melting snowpack; or}
\]

\[
= 0.0, \quad \text{For a frozen Snowpack}
\]  \hspace{1cm} (D.2)

\(^1\)See Chap. 6.
where $\alpha$, $\beta$ and $d_m$ are as defined in section 6.6.1.

Assuming that the snowpack is either frozen or partially melting\(^2\), and substituting Eq. D.2 into Eq. D.1:

$$Q_s(t) = \sum_{i=1}^{k} [\gamma(t+1-i)\alpha(t+1-i)C(t+1-i) + \beta(t+1-i)]q(i)$$

$$= \sum_{i=1}^{k} \gamma(t+1-i)\alpha(t+1-i)C(t+1-i)q(i) + \sum_{i=1}^{k} \gamma(t+1-i)\beta(t+1-i)q(i) \quad (D.3)$$

Since the rainfall is assumed negligible, the snowmelt flow component can be estimated equal to the total; observed flow less the baseflow. This “observed” snowmelt flow component is referred to as $Q_{so}$ to distinguish it from the calculated snowmelt flow component $Q_s$. It also is assumed that:

$$Q_{so}(t) = Q_s(t) + v(t) \quad (D.4)$$

where $v(t)$ is a random error of zero mean and variance $\sigma_v^2$.

Substituting Eq. D.3 into Eq. D.4 and rearranging results in:

$$Q_{so}(t) - \sum_{i=1}^{k} \gamma(t+1-i)\beta(t+1-i)q(i) = \sum_{i=1}^{k} \gamma(t+1-i)\alpha(t+1-i)C(t+1-i)q(i) + v(t) \quad (D.5)$$

Eq. D.5 constitutes the basic equation for updating the snowmelt parameters. The development of the updating model for the cloud cover factor $\eta_c$ is presented in the following section. The basic equations of the updating models for the other snowmelt parameters $\Delta T_s$ and $\eta_a$ are presented in sections D.1.2 and D.1.3, respectively.

\(^2\)This is an acceptable assumption since for a completely melted snowpack the snowmelt is insignificant.
Appendix D. Derivation of the Flow-Snowpack Updating Model

D.1.1 Updating \( \eta_c \)

Eq. D.5 can be modified to include the parameter \( \eta_c \) as a correction factor for the past “k” cloud covers, as follows:

\[
Q_{so}(t) = \sum_{i=1}^{k} \gamma(t + 1 - i)\beta(t + 1 - i)q(i) = \eta_c(t) \sum_{i=1}^{k} \gamma(t + 1 - i)\alpha(t + 1 - i)C(t + 1 - i)q(i) + v(t) \tag{D.6}
\]

Eq. D.6 can be then used to calculate the value of \( \eta_c \) that yield the exact value of the measured snowmelt flow component, as follows:

\[
\eta_c(t) = \frac{Q_{so}(t) - \sum_{i=1}^{k} \gamma(t + 1 - i)\beta(t + 1 - i)q(i)}{\sum_{i=1}^{k} \gamma(t + 1 - i)\alpha(t + 1 - i)C(t + 1 - i)q(i)} \tag{D.7}
\]

Using Eq. D.7 in updating does not take into consideration the effect of errors in the flow measurements, which can be significant in the vicinity of the high flows. To account for these errors, \( \eta_c \) is assumed to be a stochastic variable that is related to its previous value as follows:

\[
\eta_c(t) = \eta_c(t - 1) + w_c(t - 1) \tag{D.8}
\]

where \( w_c(t) \) is a random error of zero mean and variance \( \sigma^2_c \).

Eqs. D.8 and D.6 are the state and measurement equations, respectively, that constitute the state-space model. The Kalman filter can be applied on this state-space model to update the parameter \( \eta_c \) using real-time flow measurements.

D.1.2 Updating \( \Delta T_s \)

Using the same procedure applied for the parameter \( \eta_c \), the following state-space model is developed for updating the cloud cover parameter \( \Delta T_s \):

\[
\Delta T_s(t) = \Delta T_s(t - 1) + w_\Delta(t - 1) \quad \text{State Equation} \tag{D.9}
\]
Appendix D. Derivation of the Flow-Snowpack Updating Model

\[ Q_{so}(t) - \sum_{i=1}^{k} \gamma(t+1-i)\beta^*(t+1-i)q(i) = \]

\[ \frac{1}{\Delta T_s(t)} \sum_{i=1}^{k} \gamma(t+1-i)\alpha^*(t+1-i)q(i) + v(t) \quad \text{Measurement Equation (D.10)} \]

where \( w_\Delta \) is a random noise of zero mean and variance \( \sigma_\Delta^2 \); and \( \alpha^* \) and \( \beta^* \) are as defined in section 6.6.2.

D.1.3 Updating \( \eta_a \)

The state-space model for updating the snow albedo correction factor \( \eta_a \) can be expressed as follows:

\[ \eta_a(t) = \eta_a(t-1) + w_a(t-1) \quad \text{State Equation (D.11)} \]

\[ Q_{so}(t) - \sum_{i=1}^{k} \gamma(t+1-i)\beta^o(t+1-i)q(i) = \]

\[ \eta_a(t) \sum_{i=1}^{k} A(t+1-i)\gamma(t+1-i)\alpha^o(t+1-i)q(i) + v(t) \quad \text{Meas. Equation (D.12)} \]

where \( w_a \) is random noise of variance \( \sigma_a^2(t) \); and \( \alpha^o \) and \( \beta^o \) are as defined in section 6.6.3.