Midlands Catchment Runoff Model: Model Description and User Guide

A. J. Robson and R. J. Moore

Centre for
Ecology & Hydrology
NATURAL ENVIRONMENT RESEARCH COUNCIL

Version 1.0

February 2009

Acknowledgements

This guide serves to bring together relevant information on the Midlands Catchment Runoff Model (MCRM), as well as documenting the extension of the original hourly model to work at a 15 minute time-step.

The model was originally developed by Richard Douglas, with additional input from Cliff Dobson, as the Severn-Trent Catchment Runoff Model within the Severn-Trent Region of the National Rivers Authority (now Midlands Region of the Environment Agency.

This document draws heavily on the following reports, arranged in chronological order:

Harding, R.J. and Moore, R.J. 1988. Assessment of snowmelt models for use in the Severn Trent flood forecasting system. Contract report to Severn Trent Water Authority, December 1988, Institute of Hydrology, 41pp.

Wallingford Water 1994. A flood forecasting and warning system for the River Soar: Stage 2 Report. Contract report to the National Rivers Authority Severn Trent Region, Version 2.0, January 1995, 99 pp plus Appendices, Wallingford Water, Wallingford, UK (available from Centre for Ecology & Hydrology, Wallingford).

Moore, R.J., Harding, R.J., Austin, R.M., Bell, V.A. and Lewis, D.R. 1996. Development of improved methods for snowmelt forecasting. R&D Note 402, Research Contractor: Institute of Hydrology, National Rivers Authority, 192pp.

Moore, R.J and Bell, V.A. 2001. Comparison of rainfall-runoff models for flood forecasting. Part 1: Literature review of models. Environment Agency R&D Technical Report W241, Research Contractor: Institute of Hydrology, September 2001, Environment Agency, 94pp.

Delft Hydraulics and Tessella 2005. National Flood Forecasting System. MCRM Module and Adapter User and System Documentation. Version 1.1, Report prepared for the Environment Agency.

Centre for Ecology and Hydrology
Maclean Building
Crowmarsh Gifford
Wallingford
Oxfordshire
OX10 8BB
UK
Telephone +44 (0) 1491 838800
Main Fax +44 (0) 1491 6924241
Email: ForecastModelSupport@ceh.ac.uk

Centre for Ecology & Hydrology
NATURAL ENVIRONMENT RESEARCH COUNCIL

CONTENTS

List of Tables

List of Figures

1 Introduction

The Midlands Catchment Runoff Model (MCRM) is a classical conceptual rainfall-runoff model based on continuous soil moisture accounting principles. The model was developed in the Midlands Region of the Environment Agency and applied to catchments across the Severn and Trent river basins. It is the rainfall-runoff model used in the Midlands Flood Forecasting System (MFFS).

The MCRM was originally developed to run at an hourly time-step. It has now been converted to also run at shorter time-steps, with the primary aim being for use in flood forecasting systems operating at a 15 minute time-step. Although this version of the model supports time-steps of 60, 30, 20, 15, 10 and 5 minutes, the 15 minute time-step is seen as the primary time-step of operational interest. Additional consideration has been given to the error forecast model component at the 15 time-step to make the results optional. For the other short time-steps, a slightly less accurate approximation is used.

This document serves two main purposes. First, it provides an up-to-date description of the model taking account of the variable time-step model formulation. Second, it serves as a User Guide to the model software in NFFS module adapter form, providing details of the input and output files involved.

Section 2 contains a detailed description of the rainfall-runoff model along with its snowmelt, reservoir and error forecast components. Section 3 provides a guide to the software in module adapter form, detailing the input and output data files required in running the model.

2 MCRM Model Description

2.1 Overview

The rainfall-runoff model MCRM is based on classical conceptual water storage accounting principles applied to the land (soil- and ground-water) and river channels. An outline of the model, previously known as the Severn-Trent Catchment Runoff Model, is provided by Bailey and Dobson (1981), Wallingford Water (1994) and Moore and Bell (2001). Reviews of the snowmelt component are given in Harding and Moore (1988) and Moore *et al.* (1996). A schematic of the rainfall-runoff model structure is shown in Figure 2.1. This figure omits the snowmelt, reservoir and error prediction components of the overall model which are outlined later.

The model comprises three main stores: an interception store, a soil moisture store and a groundwater store. Rapid runoff is generated from the soil moisture store, the proportion of the input to the store becoming runoff increasing exponentially with decreasing soil moisture deficit. "Percolation" to the groundwater store occurs when the soil is supersaturated, increasing as a linear function of the negative deficit. When super-saturation exceeds a critical value, "rapid drainage" also occurs as a power function of the negative deficit in excess of the critical value (the so-called excess water). This rapid drainage along with rapid runoff forms the soil store runoff.

Evaporation occurs preferentially from the interception store at a rate which is a fixed proportion of the catchment potential evaporation. A proportion of any residual evaporation demand is then met by water in the soil store, the proportion varying as a function of the soil moisture deficit. Drainage of the groundwater store to baseflow varies as a power function of water in storage, the exponent being fixed at 1.5.

The total output, made up of baseflow and soil store runoff, is then lagged and spread evenly over a specified duration to represent the effect of translation of water from the ground to the catchment outlet. Finally, the flow is smoothed using two nonlinear storage functions, one for routing in-bank flow and the other out-of-bank flow, the two components being summed to give the catchment model outflow.

The more detailed operation of each component of the Midlands Catchment Runoff Model is described next.

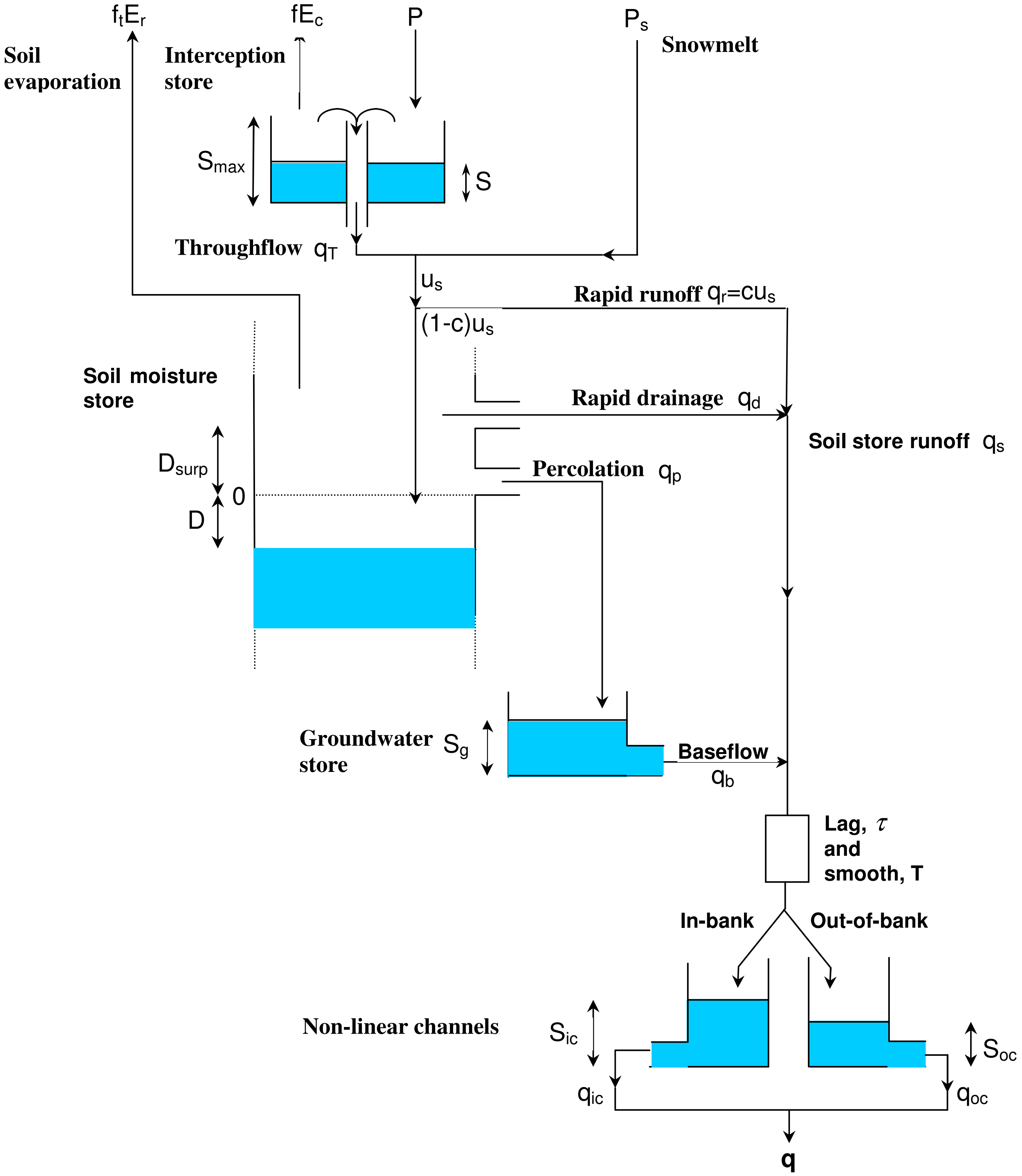


Figure 2.1 The Midlands Catchment Runoff Model

2.2 Rainfall-Runoff Model

2.2.1 Interception Store

The interception store operates as a simple bucket having a capacity, $S_{\max}$, and with water in storage, S, increasing through the addition of rainwater, P, until full when overflows, q_T, enter the soil store as throughflow. A proportion, f, of the catchment atmospheric demand for evaporation, E_c, referred to here as the catchment evaporation, is met by water in the interception store, or by a lesser amount if storage S is not sufficient. Thus we have the following sequential water balance operations for the interception store (dropping time suffixes for simplicity):

Interception Storage $S = S + P$ (2.1)

Throughflow
$$q_T = \begin{cases} S - S_{\max} & S > S_{\max} \\ 0 & otherwise \end{cases} \tag{2.2}$$

Potential interception evaporation $E_p = f\ E_c$ (2.3)

Residual evaporation demand
$$E_r = \begin{cases} E_p - \dfrac{S}{f} & E_p > S > 0 \\ 0 & E_p \le S \le 0 \\ E_c & S \le 0 \end{cases}$$
$$S = \begin{cases} 0 & E_p > S > 0 \\ S - E_p & E_p \le S;\ S > 0\,. \end{cases} \tag{2.4}$$

2.2.2 Soil store

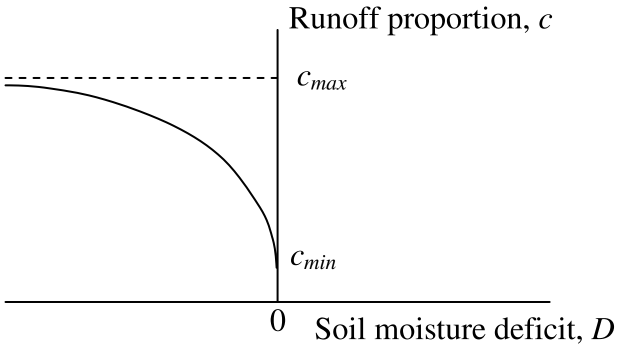
The soil store has no defined capacity, calculations proceeding on the basis of the amount of water in deficit, D. Input to the soil store, u_s, is made up of throughflow, q_T, from the interception store plus any melt, P_s, from the snowmelt component, so that $u_s = q_T + P_s$. A proportion of the input, c, does not enter the store but forms rapid runoff. This proportion increases as an exponential function of the negative deficit, from a minimum value c_0 up to a maximum value $c_{\max}$ under the control of parameter c_1 (Figure 2.2a). Thus

Rapid runoff proportion $c = min(c_{max}, c_0 \exp(-c_1 D))$ (2.5)

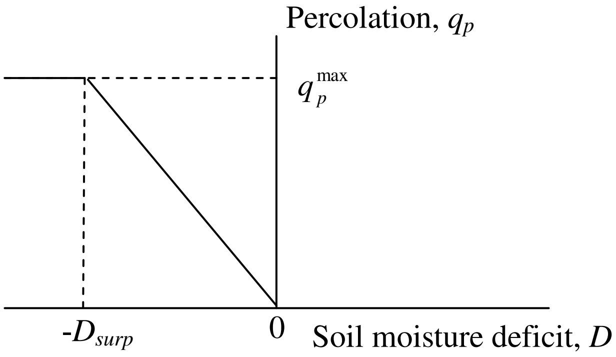
Rapid runoff $q_r = cu_s$ (2.6)

Soil moisture deficit $D = D - (1 - c)u_s$. (2.7)

(a) Rapid runoff proportion as a function of negative soil moisture deficit (moisture surplus)



(b) Percolation to groundwater as a function of negative soil moisture deficit (moisture surplus)



(c) Rapid drainage as a function of (negative) soil moisture deficit, D (or excess water, $W = -(D_{surp} + D)$)

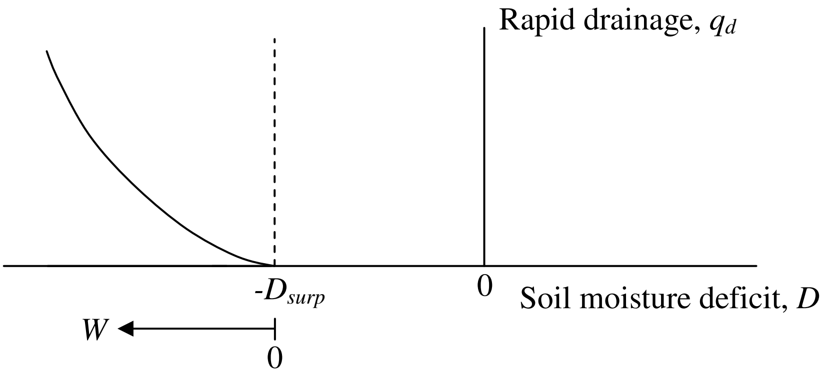


Figure 2.2 Rapid runoff, percolation and rapid drainage functions in the Midlands Catchment Runoff Model.

In practice the calculation is carried out incrementally, for each unit of input u_s, and the q_r values summed to account more accurately for the nonlinear dependence of the runoff proportion on the negative deficit.

Percolation to groundwater occurs only for negative deficits ($D < 0$) when it is governed by the equation

$$q_P = \begin{cases} \dfrac{\Delta t q_p^{\max} D}{-D_{surp}} & -D_{surp} < D \leq 0 \\ \Delta t q_p^{\max} & D \leq -D_{surp} \end{cases} \tag{2.8}$$

where $q_p^{\max}$ is the maximum percolation rate parameter and D_{surp} is the soil store moisture surplus parameter, Δt is the model time-step in hours, and where $q_p^{\max} \leq D_{surp}$. Figure 2.2b illustrates the form of the percolation function. The deficit is updated using

$$D = D + q_p. \tag{2.9}$$

Rapid drainage, q_d, which like rapid runoff bypasses the groundwater store, is generated when the negative deficit exceeds a critical value, D_{surp}, and gives rise to "excess water" conditions. Excess water is given by

$$W = -\left(D_{surp} + D\right) \qquad D < -D_{surp} \tag{2.10}$$

and rapid drainage is governed by the power function

$$q_d = \frac{\Delta t W^{\gamma_d}}{k_d} \tag{2.11}$$

where γ_d is a soil function exponent and k_d is a soil function coefficient. Figure 2.2c illustrates the form of the rapid drainage function. The deficit is updated using

$$D = D + q_d. \tag{2.12}$$

Soil store runoff, q_s, is then

$$q_s = q_r + q_d. \tag{2.13}$$

Finally, the soil store is further depleted by any residual evaporation demand, E_r, according to the soil water evaporation function which gives soil evaporation as

$$E_s = f_t E_r \tag{2.14}$$

where the transpiration factor, f_t, is given by

$$f_t = \begin{cases} T_p & D < E^D_{max} \\ T_m & D > E^D_{\min} \\ T_P - \dfrac{\left(D - E^D_{\max}\right)\left(T_p - T_m\right)}{E^D_{\min} - E^D_{\max}} & otherwise \end{cases} \tag{2.15}$$

with $T_m \le f_t \le T_p$. Here T_p and T_m are the potential and minimum transpiration parameters and E^D_{max} and E^D_{min} the corresponding deficit values at which these limiting conditions first apply.

Finally, the soil moisture deficit is updated using

$$D = D + E_s \ . \tag{2.16}$$

2.2.3 Groundwater store

The groundwater store behaves as a nonlinear storage with an exponent value of 1.5. It receives percolation, q_p, from the soil moisture store as input and output is baseflow, q_b. The groundwater storage is updated according to

$$S_g = S_g + q_p \qquad q_p > 0 \tag{2.17}$$

and baseflow is given by the storage function

$$q_b = \frac{\Delta t S_g^{1.5}}{1000 K_g} \tag{2.18}$$

with parameter K_g. Adjustment to the storage then follows as

$$S_g = \begin{cases} S_g - q_b & S_g > 0 \\ 0 & otherwise. \end{cases} \tag{2.19}$$

2.2.4 Lag and spread of catchment runoff

The total runoff is the sum of baseflow and soil store runoff, $q_b + q_s$. This is lagged by a fixed time interval, τ, and spread evenly over a specified duration, T, in order to represent the translation of water from the ground to the catchment outlet.

2.2.5 Nonlinear smoothing of catchment runoff

The last operation in the Midlands Catchment Runoff Model is the application of a nonlinear smoothing function to produce a smooth catchment outflow hydrograph. Nonlinear storage functions are used for in-bank and out-of-bank flows, which are treated separately as follows. After first adding the lagged runoff to the in-channel storage, S_{ic}, the out-of-bank component of the input and in-channel storage are calculated as

$$u_{oc} = \begin{cases} S_{ic} - S_{bf} & S_{ic} > S_{bf} \\ 0 & otherwise \end{cases} \tag{2.20}$$

$$S_{ic} = S_{bf} \qquad\qquad S_{ic} > S_{bf}\,.$$

The in-channel outflow is given by the nonlinear storage function

$$q_{ic} = \begin{cases} \Delta t k_{cr} S_{ic}^{\gamma_{cr}} & q_{ic} \le .75 \Delta t S_{ic} \\ .75 \Delta t S_{ic} & otherwise \end{cases} \tag{2.21}$$

where k_{cr} and γ_{cr} are the in-channel routing coefficient and exponent. A similar expression is used to obtain the out-of-bank outflow, q_{oc}, from the out-of-bank storage, S_{oc}. Updating of the in-channel storage follows

$$S_{ic} = S_{ic} - q_{ic} \tag{2.22}$$

and for the out-of-channel store

$$S_{oc} = S_{oc} + u_{oc}\,. \tag{2.23}$$

Finally, the total catchment outflow is calculated as

$$q = q_{ic} + q_{oc}\,. \tag{2.24}$$

2.2.6 Model Parameters

A summary of the model parameters used in the Midlands Catchment Runoff Model is presented in Table 2.1 together with the units used in the model.

Table 2.1 Parameters in the Midlands Catchment Runoff Model.

Parameter	Unit	Description
f_c	none	Rainfall factor
$S_{\max}$	mm	Capacity of interception store
f	none	Fraction of catchment evaporation potentially met by interception storage
c_0	none	Minimum value of rapid runoff proportion
c_1	mm^{-1}	Parameter in rapid runoff proportion function
$c_{\max}$	none	Maximum value of rapid runoff proportion
q_p^{max}	mm h^{-1}	Maximum percolation rate
D_{surp}	mm	Maximum soil store moisture surplus
γ_d	none	Soil function exponent controlling rapid drainage
k_d	$\mathrm{h\ mm}^{\gamma_d-1}$	Soil function coefficient controlling rapid drainage
T_p	none	Potential transpiration factor
T_m	none	Minimum transpiration factor
E_{max}^D	mm	Deficit below which potential transpiration factor applies
E_{min}^D	mm	Deficit above which minimum transpiration factor applies
K_g	h mm$^{0.5}$	Time constant in baseflow storage function
τ	h	Time lag applied to total runoff
T	h	Duration of time spread applied to total runoff
S_{bf}	mm	Channel storage at bankfull
k_{cr}	$\mathrm{h}^{-1}\mathrm{mm}^{1-\gamma_{cr}}$	In-channel routing storage coefficient
γ_{cr}	none	In-channel routing storage exponent
k_{or}	$\mathrm{h}^{-1}\mathrm{mm}^{1-\gamma_{or}}$	Out-of-bank channel routing storage coefficient
γ_{or}	none	Out-of-bank channel routing storage exponent
Δt	h	Time-step in hours (e.g. 0.25 for 15 min data)

2.3 Snowmelt Model

The Midlands Catchment Runoff Model incorporates a simple snowmelt component which estimates the snow input from heated raingauge and air temperature measurements. Changes of snow density, snowmelt and routing of melt through the snowpack are computed by the model at an hourly time interval. Manual state updating of the snow depths and densities using field measurements are occasionally made. The Snowmelt Model is based on the temperature index method and also incorporates a mechanical snowpack compaction relation. A set of four relations control the detailed operation of the model, and these are summarised below.

2.3.1 Mechanical snowpack compaction

Given a snowpack depth, d, this is decreased by a proportion, c, during the model time interval, Δt (measured in hours). The compaction coefficient, c, is dependent on air temperature, T, and bounded, such that

$$c^* = .996 - 0.002T$$

and

$$c = \begin{cases} .998^{\Delta t} & c^* > .998 \\ \left(c^*\right)^{\Delta t} & .996 \le c^* \le .998 \\ .996^{\Delta t} & c^* < .996 \end{cases} \tag{2.25}$$

Thus the pack is compacted at a rate of between 0.2 and 0.4% per hour, increasing linearly with temperature over the range -1°C to 0°C. This model component is implemented first, essentially compacting the snow depth remaining at the end of the previous model time-step.

2.3.2 Snow/rain threshold

Raingauges are assumed to be heated raingauges which melt any snow falling into the funnel. Measured rainfall is assigned as snow if the air temperature is below a critical temperature, T_c. Whilst T_c is a model parameter, and therefore variable, it is currently set at 2.5°C.

2.3.3 Density of fresh snow

The density of fresh snow, ρ_s, is taken to be controlled by the temperature dependence relation

$$\rho_s = \begin{cases} 0.1 + 0.015T & \rho_s \ge \rho_c \\ \rho_c & \text{otherwise} \end{cases} \tag{2.26}$$

where ρ_c is the minimum permissible density, currently set at 0.07 (or 7%) and corresponding to a temperature of -2°C.

2.3.4 Melt equation

The potential melt equation used is of simple temperature index form, but extended to include an exponent. Thus the potential melt, M_p, is given by

$$M_p = \begin{cases} \Delta t\, f\, (T - T_m)^x & T > T_m \\ 0 & \text{otherwise.} \end{cases} \tag{2.27}$$

The temperature factor, f, and the temperature exponent, x, are currently set at 0.1 and 1.2 respectively, and M_p has units mm hr^{-1}. If the air temperature is below the threshold snowmelt temperature, T_m (currently set at 0.5°C) then the potential melt is zero.

2.3.5 Snow depth and water equivalent accounting

Consider an interval when an amount, P, of snow falls on a snowpack of depth, d mm, of water equivalent, S mm water. Then d and S are updated as follows:

$$S^* = S + P \tag{2.28}$$

$$d^* = d + \frac{P}{\rho_s} \tag{2.29}$$

with ρ_s calculated according to (2.26). The snowpack density is then

$$\rho = \frac{S^*}{d^*}. \tag{2.30}$$

A similar procedure is followed if P falls as rain, but the depth of snowpack is held fixed, and only the pack's water equivalent depth, S, is increased.

2.3.6 Ripening, melt and drainage of the snowpack

If snowmelt occurs ($T > T_m$) during the interval then a further adjustment to the depth and water equivalent of the pack is required. If the potential melt, M_p, calculated according to equation (2.27) exceeds the pack water equivalent, S^*, then the pack disappears, relevant quantities are re-zeroed (d, S), and the actual melt, M, set to S^*. However, if some of the pack remains following melt then the depth is reduced so that

$$d^{**} = \max\left(0.01, d^{*} - \frac{M_p}{\rho}\right) \tag{2.31}$$

and the new pack density calculated, assuming that the melt is retained in the pack (i.e. S^{*} does not change), so

$$\rho^{*} = \frac{S^{*}}{d^{**}}. \tag{2.32}$$

If the pack density remains below the threshold density, ρ_m, then no melt is assumed to occur ($M = 0$): ρ_m has a value 0.35 or 35%. However, if ρ_m is exceeded then the maximum water equivalent of the pack is calculated as the product of the depth and threshold density, and any water in excess of this is released from the pack i.e.

$$M = S^{*} - d^{**}\,\rho_m. \tag{2.33}$$

Then the pack's water equivalent, depth and density are recomputed as follows

$$S^{**} = S^{*} - M, \tag{2.34}$$

$$d^{***} = d^{**} - M, \tag{2.35}$$

since the density of the released melt is unity, and

$$\rho^{**} = \frac{S^{**}}{d^{***}}. \tag{2.36}$$

Thus melt is absorbed by the pack until the pack density exceeds a critical threshold value, ρ_m, when melt is released from the pack and is input to the rainfall-runoff model. Note that the concept of ρ_m is similar to that of a maximum LWC (Liquid Water Content) except the threshold here is liquid plus solid density and can allow very high and possibly unrealistic liquid water contents.

Table 2.2 provides a summary of the model parameters included in the Midlands Snowmelt Model formulation. Note that Harding and Moore (1988) suggest that a value for the critical density of a ripe pack, ρ_m, would be near 13.5%, giving a critical liquid water content of 10% (and not 70% for $\rho_m = 0.35$) if the pack water equivalent is 100 mm.

Table 2.2 The Midlands Snowmelt Model parameters

Parameter	Meaning	Typical value	Unit
T_c	Maximum temperature for snowfall	2.5	°C
ρ_s	Minimum density of snowfall (fresh snow)	0.07	mm (mm water)$^{-1}$
T_m	Minimum temperature for snowmelt	0.5	°C
f	Melt equation temperature factor	0.1	mm hr^{-1}(°C)$^{-1}$
x	Melt equation temperature exponent	1.2	dimensionless
ρ_m	Density of ripe snowpack	0.35	mm (mm water)$^{-1}$

2.4 Reservoir Balance Model

The MCRM can incorporate the effect of a reservoir within the catchment to be modelled if required. Reservoir outflow comprises the sum of the controlled release Q_{rel} and any reservoir spill Q_{sp}. With p_r the % proportion of the catchment outflow coming from the reservoir, the modelled inflow to the reservoir is a fraction $p_r/100$ of the flow from the rainfall-runoff model part of the MCRM, Q_{in}. This construct allows the reservoir to be either located at the outlet of the catchment ($p_r = 100$) or at an interior location ($p_r < 100$). Abstractions from the reservoir are made at a rate q_{abs}.

If reservoir level, H_r, is available then the reservoir storage, S_r, is directly estimated from this using the reservoir rating (storage-level) equations. These equations take the form

$$S_r = c_r(H_r - d_r)^{\gamma_r} \qquad (2.37)$$

where c_r, d_r and γ_r are the reservoir rating coefficient, datum correction and exponent parameters respectively. The parameter values are set for each of n_r sections defining the rating and apply up to the upper limit of the section H_r^u.

If the reservoir level is not observed then, the reservoir storage is calculated from mass balance as

$$S_r = S_r - \frac{\Delta t\, q_{abs}}{24} + 3.6\Delta t\left(\frac{p_r}{100}Q_{in} - Q_{out}\right). \qquad (2.38)$$

Here, Q_{out} is either the observed flow, Q_{obs} (provided that $p_r = 100$), or the last reservoir release Q_{rel}.

Reservoir spill occurs if the reservoir level exceeds zero. The reservoir level is either observed or estimated from the reservoir storage S_r. The rate of spill, Q_{sp}, is then calculated as

$$Q_{sp} = c_{sp} H_r^{\lambda_{sp}} . \tag{2.39}$$

However, if $p_r = 100$ and there is observed output flow, then if the spill is greater than the observed output flow it is set to

$$Q_{sp} = Q_{obs} - q_{rel}^* / (24 \times 3.6) \tag{2.40}$$

where q_{rel}^* is the default reservoir release rate.

Note that if neither output flow nor reservoir level have been observed (e.g. during forecast mode) and the reservoir storage is above capacity, then the reservoir storage, level and spill are updated using a 1 minute time-step to give improved accuracy.

The controlled reservoir release, Q_{rel}, remains the same unless there is observed flow and $p_r = 100$, in which case

$$Q_{rel} = Q_{obs} - Q_{sp} . \tag{2.41}$$

Note that if $p_r < 100$ then Q_{rel} will always remain fixed at the value set in the states file (see Section 3.1.7) for a given reservoir: it cannot be set via the parameter file (see Section 3.1.5). For these reasons it is recommended that the Reservoir Balance Model is only used when the reservoir is located at the catchment outlet.

The total outflow from the catchment is given by

$$Q = \left(1 - \frac{p_r}{100}\right) Q_{in} + Q_{rel} + Q_{sp} \tag{2.42}$$

Table 2.3 Reservoir Balance Model parameters

Parameter	Meaning	Typical value	Unit
p_r	Proportion of output flow coming from reservoir	100	%
$S_r^{\max}$	Reservoir maximum storage capacity	40000	Ml
q_{abs}	Reservoir abstraction rate	0	Ml day^{-1}
q_{rel}^*	Default reservoir release rate	100	Ml day^{-1}
c_{sp}	Reservoir spill coefficient	225	$\text{s}^{-1}\text{m}^{3-\gamma_{sp}}$
γ_{sp}	Reservoir spill exponent	1.5	dimensionless

2.5 Error Forecast Model

The Environment Agency Midlands Region approach to forecast updating is referred to as the "Error Forecast Model". It examines the difference between observed and simulated outflows over the last m time-steps of the hindcast period (usually equating to 6 hours). A judgement is made on how predictable future errors are and forecast outflows are adjusted accordingly. The approach is described in greater detail below.

First the observed errors over the last m time-steps are isolated to form the set $\{\eta_{-m+1},...,\eta_0\}$ where η_0 is the most recent observed error and

$$\eta_i = Q_i - q_i \qquad (2.43)$$

where Q_i and q_i are the observed and simulated flows at instant i. The error differences are then calculated as

$$\delta_i = \eta_i - \eta_{i-1} \quad i = -m+2, -m+3, ..., -1, 0 \qquad (2.44)$$

and from this set $\{\delta_i\}$ the maximum error difference is calculated as

$$\delta_{\max} = \max\{|\delta_i|\}. \qquad (2.45)$$

The mean observed discharge is calculated as

$$\overline{Q} = \frac{\sum_{i=0}^{i=-m+1} |Q_i|}{m}. \qquad (2.46)$$

Using $\overline{Q}$ and $\delta_{\max}$ the duration in the forecast period over which "normal" updating will be undertaken is defined as

$$\tau = \frac{\overline{Q}}{2\delta_{\max}}. \qquad (2.47)$$

Thus, the duration for normal updating is shortened with larger errors, when judged in proportion to the mean observed discharge.

Consider now the forecast period of duration τ. An autoregressive model for the error differences is used to provide estimates of the next error change, $\hat{\delta}_i$. This takes the form

$$\hat{\delta}_i = \phi_1 \delta_{i-1} + \phi_2 \delta_{i-2} + ... + \phi_{m-1} \delta_{i-m+1} \quad i = 1, 2, ..., \tau \qquad (2.48)$$

which is computed recursively, substituting $\hat{\delta}_i$ .on the right-hand side when actual δ_i values are unavailable for future times. The autoregressive coefficients $\{\phi_j, j = 1, 2, ..., m-1\}$ are interpreted as a set of error reduction factors. In practice for hourly data $m = 6$ and $\{\phi_j\} = (.36,.21,.12,.07,.04)$, which implies $\phi_j = .36(.583)^{j-1}$, $j = 1,2,...,m-1$. That is the factors decrease exponentially into the past.

For other time-steps the following autoregressive coefficients are used

$$m = 6/\Delta t \tag{2.49}$$

$$\phi_j = \alpha\, 0.583^{(m-1-j)\Delta t} \tag{2.50}$$

where Δt is the time-step as a fraction of hours, and α is a constant that depends on the time-step. For time-steps shorter than 15 minutes, the same set of autoregressive constants are used as for 15 minutes.

For values of Δt other than 15 minutes, the following estimate of α is used:

$$\alpha = 0.36 \frac{(1-0.583^{\Delta t})}{(1-0.583)}. \tag{2.51}$$

This value of α will give a slightly faster decay in the series for time-steps shorter than 1 hour. A more accurate solution would involve solving complex polynomial equations.

The main time-step of interest (other than hourly) is 15 minutes ($\Delta t = 0.25$), so further consideration for this time-step has been undertaken. The above formula gives the result $\alpha = 0.109$ and is considered a lower bound. It can also be shown that a likely upper bound for α is 0.117. This value would give a recession slower than the hourly series. Since there is no knowledge of the reasons for choosing the original hourly parameter values, the rounded intermediate value of 0.11 has been selected for this time-step.

The error predictor is then formulated as the sum of the previous error and the predicted error change

$$\hat{\eta}_i = \eta_{i-1} + \hat{\delta}_i \quad i = 1,2,...,\tau \tag{2.52}$$

which is again computed recursively with $\eta_0 = Q_0 - q_0$ and thereafter substituting $\hat{n}_{i-1}$ for n_{i-1} on the right hand side.

The updated flow forecast is then simply given by the sum of the simulation forecast and the predicted error:

$$\hat{q}_i = q_i + \hat{\eta}_i \quad i = 1,2,...,\tau\,. \tag{2.53}$$

In practice if the updated forecast is negative, the forecast is reset to the forecast at the previous time-step.

This updating scheme is therefore based on forecasting the error differences, using an exponentially-fading weighted average of $m-1$ past observed and/or forecast error differences. The forecast error is then the old error plus the forecast error difference. Adding this to the simulation model forecast gives the required updated forecast.

The updating procedure for the period $\tau + 1$ to 2τ, asymptotes the updated forecasts back to the raw forecast at a lead time of 2τ. A "final error divider" is defined as $\Delta = \eta_t / \tau$. This is used in the following recursions for $i = \tau + 1, \tau + 2, ..., 2\tau$ to obtain the future forecast errors and updated flow forecasts:

$$\hat{\eta}_i = \hat{\eta}_{i-1} - \Delta \tag{2.54}$$

$$\hat{q}_i = q_i + \hat{\eta}_i . \tag{2.55}$$

Again if the updated forecast is negative the forecast reverts to that at the previous time-step. In the event that missing data occur in the hindcast period no updating is attempted and the simulated flows are used over the whole forecast period.

3 Using the MCRM Module

The MCRM module consists of a single executable called `mcrm.exe`. This executable, `mcrm.exe` is activated directly by the General Adapter. Command line arguments that are passed to the executable in conjunction with the working directory define which model is to be run.

The MCRM module runs for one catchment at a time. A simplified flowchart indicating the module operation is shown in Figure 3.1.

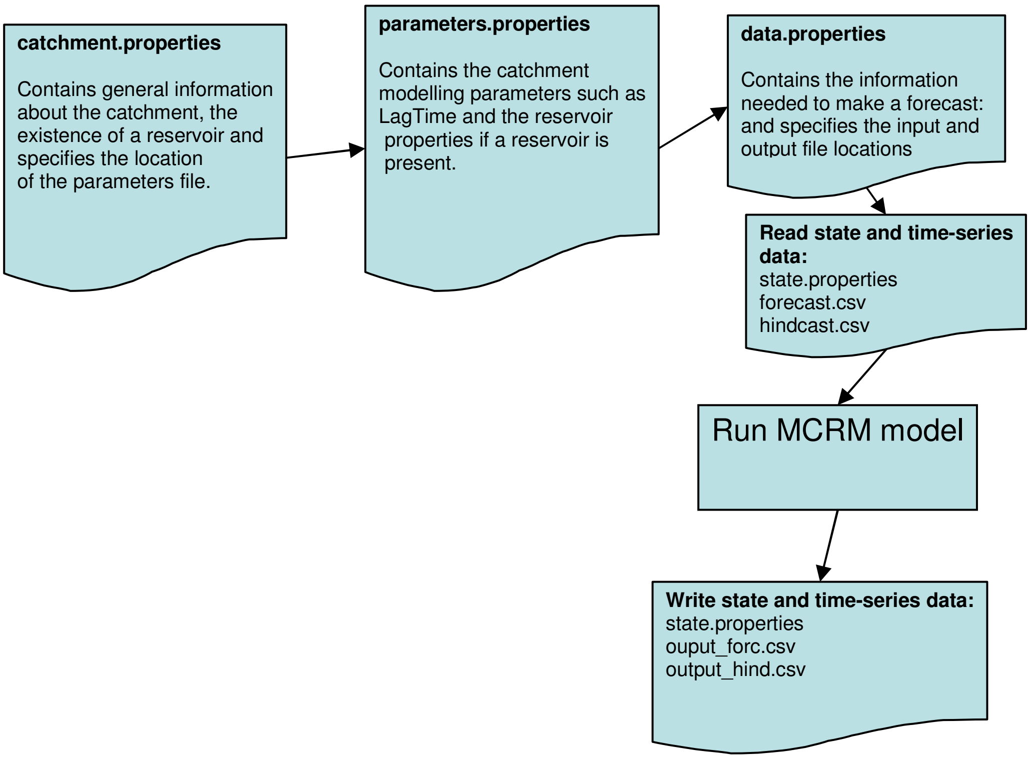


Figure 3.1 Data flow chart for the MCRM module

The module runs using an initial state as input for a specified number of hours (backhours) in the hindcast period. After this period the model state is saved and the model continues to run for the forecast period (forchours). No state information is saved at the end of the forecast period. The time-step of the hindcast period is identical to the first time-step of the forecast period.

By default the module will run an error model if observed discharge is available. This option can be switched off using a setting in the `catchment.properties` file (see Section 3.1.3).

3.1 MCRM Input data requirements

3.1.1 Overview

The MCRM module has the following input data file requirements:

- *Command line arguments*. MCRM requires two command line arguments. The first argument specifies the location and name of the `catchment.properties` file while the second argument specifies the name and the location of the `data.properties` file.
- *Configuration files:* For each catchment a configuration file (`catchment.properties`) . This file contains general information about the catchment and specifies the location of the parameters file. `data.properties`. This file specifies the locations of the input and output time-series and state values as well as the length of the hindcast and forecast period.
- *Parameter file*. The `parameters.properties` file contains all the parameters for one catchment. It also includes the reservoir ratings if a reservoir is present. If a time-step shorter than 1 hour is needed this is specified in the parameter file.
- *Input data files*. The module requires two .csv files containing input time-series: one file with hindcast data and one file with forecast data. The locations of these files are specified in the `data.properties` file.
- *Initial state file*. The initial state for a catchment is taken from the `state.properties` file. The location of this file is specified in the `data.properties` file.
- MCRM regards any value < -9999.999 as a missing value. To avoid confusion, all MCRM in- and output should use -10000 as the missing value definition.

Within the NFFS the `catchment.properties` and `data.properties` will normally be generated by the MCRM module adapter. The `parameters.properties` is generally static (in the on-line system) but can be edited using any ASCII editor (such as Notepad).

3.1.2 Command line arguments

The MCRM module needs a number of parameters to find its configuration files. The first argument specifies the location and name of the `catchment.properties` file while the second argument specifies the name and the location of the `data.properties` file. The general syntax of the command line arguments is specified below:

```
MCRM.exe catchment.propertiesfile data.propertiesfile
```

Bold items are literal items whilst italics items should be replaced by a valid filename.

Command-line item	Description
MCRM.exe	The name of the MCRM executable
catchment.propertiesfile	Full path to the catchment.properties file (see section 3.1.3)
data.propertiesfile	Full path to the data.properties file (see section 3.1.4)

3.1.3 Catchment.properties file

The `catchment.properties` file is an ASCII text file divided in sections. Each section can have a number of variables with associated values separated by an equal sign ("="). The variable names are case insensitive. If the variable is a file name, the case of the name is retained: this can be significant on operating systems that have a case-sensitive file system. Comment lines are allowed and are prefixed by a sharp (#) sign. An example is given below:

```
[General]
Name = HAFREN
Reservoir = yes
Area = 61.3
UseError=NO
parameters=parameters.properties
```

The following sections are required in this file:

```
[General]
```

This section must hold the following variables:

Variable	description	Type	Required
Name	Name of the catchment	String	YES
Reservoir	Indicates presence of a reservoir	YES or NO	YES
Area	Area of the catchment	real	YES
UseError	Indicates the use of the error module if observed discharge is present.	YES or NO	YES
Parameters	Location and name of the parameters.properties file	String	YES

3.1.4 Data.properties file

The `data.properties` file is an ASCII text file divided into sections. Each section can have a number of variables with associated values separated by an equal sign. The variable names are case insensitive. If the variable is a file name, the case of the name is retained: this can be significant on operating systems that have a case-sensitive file system. Comment lines are allowed and are prefixed by a sharp (#) sign. An example is given below:

```
[Forecast]
backhours=12
forchours=50
logfilelocation=..\output\mcrm.log

[Timeseries]
#input time series
hindcast=..\input\testhind.csv
forecast=..\input\testfor.csv
#output time series
outhindcast=..\output\out_hind.csv
outforecast=..\output\out_forc.csv
```

```
[State]
inname=..\input\hafren.state
outname=..\output\hafren.state
```

The following sections are required in this file:

```
[Forecast]

[TimeSeries]

[State]
```

The following tables list the variables that must be present in each section:

`[Forecast]`			
Variable	**description**	**Type**	**Required**
backhours	Number of lines of data provided in the hindcast input timeseries file	integer	YES
forchours	Number of lines of data provided in the forecast input timeseries file	integer	YES
logfilelocation	location and name of the mcrm log file	string	YES

Note the parameters backhours and forchours are misleadingly named. This has occurred for historical reasons: retention of the original parameter names was required in order to main backwards compatibility.

The backhours is the number of lines (time-steps) in the incoming hindcast file. The forchours is the number of lines (time-steps) in the incoming forecast file. Note also that the output hindcast and forecast data files contain one additional line to the incoming files: the extra line is at the start of the file. For the forecast file, this corresponds to time 0 and is in common with the final line of the hindcast file. For the hindcast file, the initial line is either the initial start state taken from the states file or is a start-up line with non-valid data.

`[TimeSeries]`			
Variable	**description**	**Type**	**Required**
hindcast	csv file with hindcast input time-series	string	YES
forecast	csv file with forecast input time-series	string	YES
outhindcast	csv file with hindcast output time-series	string	YES
outforecast	csv file with forecast output time-series	string	YES

`[State]`			
Variable	**description**	**Type**	**Required**
inname	name of the input state file (state.properties, see 3.1.7)	string	YES
outname	name of the input state file (state.properties, see 3.1.7)	string	YES

3.1.5 Parameter file

The `parameters.properties` file is an ASCII text file divided into sections. Each section can have a number of variables with associated values separated by an equal sign. The variable names are case insensitive. If the variable is a file name, the case of the name is retained: this can be significant on operating systems that have a case-sensitive file system. Comment lines are allowed and are prefixed by a sharp (#) sign. An example is given below:

```
[Parameters]
MaxTempSnowfall=0.0
MinDensSnowfall=0.07
MinTempSnowmelt=0.1
SnowmeltCoeff=0.1
SnowMeltExp=1.2
SnowPackRipe=0.35
IntStoreCap=1.6
IntStoreEvapFac=2.0
PotTranspFac=0.66
MaxSMDPotEvap=20.0
SMDMinTransp=400.0
MinTranspFac=0.0
RunoffCoeff=0.45
RunoffExp=0.04
MaxRunoffPerc=0.75
SSMsurplus=12.8
SFuncCoeff=15.0
SFuncExp=1.5
MaxPercolnRate=0.4
BaseFlowCoeff=2.5
LagTime=0.5
DurationResp=3.0
ChanRoutCoeff=0.1
ChanRoutExp=1.5
FPRoutFactor=0.04
FPRoutExp=1.5
BankFullFlow=30.0
ReturnFlow=30.0
StatStoreCap=0
StatEvacCoeff=0.001
StatEvacExp=1

#only needed for timesteps other than 60 minutes
[timestepinformation]
timestepinminutes=15

# the reservoir section is only needed if a
# reservoir is present (see catchment.properties)
[Reservoir]
ResGgeProp = 12
ResCap=  12
ResAbs= 12
ResRel= 12
ResSpillCoeff= 12
ResSpillExp= 0.3

#For up to six
# reservoir
# levels a rating can be defined
[reslevel_1]
ResRatCoeff=0.1
```

```
ResRatCorr=12
ResRatExp=0.3
ResRatLimit=12
```

The following sections are required in this file:

```
[Parameters]
```

The following sections are optional in this file:

```
[timestepinformation]
[Reservoir]
```

Up to six reslevel_X sections may be present (X=1 to n, n≤6):

```
[reslevel_1] …[reslevel_n]
```

The following tables list the variables that must be present in each section:

[Parameters]			
Variable	**description**	**Type**	**Required**
`MaxTempSnowfall`		real	YES
`MinDensSnowfall`		real	YES
`MinTempSnowmelt`		real	YES
`SnowmeltCoeff`		real	YES
`SnowMeltExp`		real	YES
`SnowPackRipe`		real	YES
`IntStoreCap`		real	YES
`IntStoreEvapFac`		real	YES
`PotTranspFac`		real	YES
`MaxSMDPotEvap`		real	YES
`SMDMinTransp`		real	YES
`MinTranspFac`		real	YES
`RunoffCoeff`		real	YES
`RunoffExp`		real	YES
`MaxRunoffPerc`		real	YES
`SSMsurplus`		real	YES
`SFuncCoeff`		real	YES
`SFuncExp`		real	YES
`MaxPercolnRate`		real	YES
`BaseFlowCoeff`		real	YES
`LagTime`		real	YES
`DurationResp`		real	YES
`ChanRoutCoeff`		real	YES
`ChanRoutExp`		real	YES
`FPRoutFactor`		real	YES
`FPRoutExp`		real	YES
`BankFullFlow`		real	YES
`ReturnFlow`		real	YES
`StatStoreCap`		real	YES
`StatEvacCoeff`		real	YES
`StatEvacExp`		real	YES

The timestepinformation section is required if a time-step less than 1 hour is needed.

[timestepinformation]			
Variable	**description**	**Type**	**Required**
`timestepinminutes`		real	YES

The reservoir section is only required if a reservoir exists.

[Reservoir]			
Variable	**description**	**Type**	**Required**
`ResGgeProp`		real	YES
`ResCap`		real	YES
`ResAbs`		real	YES
`ResRel`		real	YES
`ResSpillCoeff`		real	YES
`ResSpillExp`		real	YES

The reslevel_n section(s) are optional but if a section exists all variables are required.

[reslevel_n]			
Variable	**Description**	**Type**	**Required**
`ResRatCoeff`		real	YES
`ResRatCorr`		real	YES
`ResRatExp`		real	YES
`ResRatLimit`		real	YES

Legal values for the MCRM parameters in the parameters file

The MCRM code does some input checking on its parameters. The following table list the legal ranges of the MCRM parameters. These ranges are particularly important when creating `Parameter.XML` files for use with the calibration module.

[Parameters]			
Variable	**Min**	**Max**	**Other requirements**
`MaxTempSnowfall`	-1.0	3.0	-
`MinDensSnowfall`	>0	1	-
`MinTempSnowmelt`	-0.5	2.0	-
`SnowmeltCoeff`	0	1.00	-
`SnowMeltExp`	0	9.99	-
`SnowPackRipe`	>0	1.00	-
`IntStoreCap`	>0	9.9	-
`IntStoreEvapFac`	>0	9.99	-
`PotTranspFac`	>0	0.99	>MinTranspFac
`MaxSMDPotEvap`	>0	500.0	<SMDMinTransp
`SMDMinTransp`	>0	500.0	>MaxSMDPotEvap
`MinTranspFac`	0	0.99	<PotTranspFac
`RunoffCoeff`	>0	0.99	≤MaxRunoffPerc
`RunoffExp`	>0	0.1	-
`MaxRunoffPerc`	>0	0.99	≥RunoffCoeff
`SSMsurplus`	>0	99.9	-
`SFuncCoeff`	>0	99.9	-

[Parameters]			
Variable	**Min**	**Max**	**Other requirements**
SFuncExp	>0	9.99	-
MaxPercolnRate	>0	9.99	-
BaseFlowCoeff	>0	99.9	-
LagTime	0	9.9	-
DurationResp	>0	99.9	-
ChanRoutCoeff	>0	0.25	-
ChanRoutExp	>0	9.99	-
FPRoutFactor	>0	0.099	-
FPRoutExp	>0	9.99	-
BankFullFlow	>0	999.9	-
ReturnFlow	>0	999.9	-
StatStoreCap	0	50000.0	-
StatEvacCoeff	>0	0.15	-
StatEvacExp	>0	9.99	-

The timestepinformation section is optional. If not specified the time-step will default to 60 minutes.

[Timestepinformation]			
Variable	**Min**	**Max**	**Other requirements**
timestepinminutes	5	60	60, 30, 20, 15, 10, 5 supported

The reservoir section is only required if a reservoir exists.

[Reservoir]			
Variable	**Min**	**Max**	**Other requirements**
ResGgeProp	>0	100.0	-
ResCap	>0	-	-
ResAbs	>0	999.9	-
ResRel	>0	1500.0	-
ResSpillCoeff	>0	999.9	-
ResSpillExp	>0	4.0	-

The reslevel_n section(s) are optional but if a section exists all variables are required.

[reslevel_n]			
Variable	**Min**	**Max**	**Other requirements**
ResRatCoeff	999.999	2500.0	-
ResRatCorr	-99.9999	99.999	ResRatLimit + ResRatCorr >= 0
ResRatExp	>0	99.9999	-
ResRatLimit	-99.999	4.999	ResRatLimit + ResRatCorr >= 0

Other minimum and maximum values imposed by the MCRM code

```
!       state variable limits
        parameter GLB_MinSMD = -40.0
        parameter GLB_MaxSMD = 200.0
        parameter GLB_MaxGroundWStore = 200.0
        parameter GLB_MaxSnowpWE = 150.0
        parameter GLB_MaxSnowpDens = 0.5
        parameter GLB_MaxSnowpDepth = 1500.0
        parameter GLB_MaxStatStoreVol = 50000000.0
        parameter GLB_MaxResStorage = 100000.0

!       details parameters
        parameter GLB_MaxRainGges = 6
        parameter GLB_MaxRivGges = 3
        parameter GLB_MaxResGges = 4
        parameter GLB_MaxClimGges = 3
        parameter GLB_MaxRatings = 6
        parameter GLB_MaxThresh = 6
        parameter GLB_MaxRadars = 2
        parameter GLB_MaxInflows = 6

        parameter GLB_MaxAreaSize = 2000.0
        parameter GLB_MaxAlt = 500.0
        parameter GLB_MaxAAR = 2500.0
        parameter GLB_MaxAAPotEvap = 1000.0
        parameter GLB_MaxLevelThreshValue = 10.0
        parameter GLB_MaxFlowThreshValue = 1000.0
        parameter GLB_MaxradarBlocks = 14
        parameter GlB_MaxRadarPoints = 200
        parameter GLB_MaxIdNum = 8999
        parameter GLB_MaxRatCoeff = 2500.0
        parameter GLB_MinRatCoeff = -999.999
        parameter GLB_MaxResRatCorr = 99.999
        parameter GLB_MinResRatCorr = -99.9999

        parameter GLB_MaxRivRatCorr = 19.99
        parameter GLB_MinRivRatCorr = -9.9999
        parameter GLB_MaxRatExp = 99.9999
        parameter GLB_MaxResRatLim = 4.999
        parameter GLB_MinResRatLim = -99.999
        parameter GLB_MaxRivRatLim = 9.999
        parameter GLB_MinRivRatLim = 0.1
        parameter GLB_MaxFlowRatLim = 999.9
        parameter GLB_MaxDirectAbs = 999.9
        parameter GLB_MaxDefRelease = 1500.0
        parameter GLB_MaxSpillCoeff = 999.9
        parameter GLB_MaxSpillExp = 4.0
        parameter GLB_MaxRivPropn = 999.0
        parameter GLB_MaxResPropn = 100.0
        parameter GLB_MaxReachLag = 20.0
```

3.1.6 Input data files

Input time-series are stored in two separate csv files, one for the forecast period and one for the hindcast period. For the hindcast period the file should contain observed discharge, observed precipitation, observed temperature, evapotranspiration and (optional) a last column with the reservoir level. For the forecast period the file should contain forecasted-precipitation, forecasted temperature and forecasted evapotranspiration.

The number of records in the hindcast file should exactly match the number of time-steps contained in the backhours period (`data.properties` file) while the number of records in the forecast file should exactly match the number of time-steps contained within the forchours period (`data.properties` file). Columns are separated by a comma (,). Comment lines are allowed and are prefixed by a sharp (#) sign. Example files for the hindcast and the forecast period are given below:

```
# This is a comment
# Hindcast period:
# the reservoir column is optional
# Q.obs, P.obs, T.obs, evaporation, reservoir-level
0.2,0,20.5,0.02,23
0.2,0,20.5,0.02,23
0.2,0,20.5,0.02,23
0.2,0,20.5,0.02,23
0.2,0,20.5,0.02,23
0.2,0,20.5,0.02,23
```

```
# This is a comment
# Forecast period:
# P.forecast, T.forecast, ET_forecast
0,20.4,0.02
0,20.4,0.02
0,20.4,0.02
0,20.4,0.02
0,20.4,0.02
0,20.4,0.02
```

3.1.7 Initial state file

The initial state for a catchment is taken from the `state.properties` file. The location of this file is specified in the `data.properties` file. The `state.properties` file is an ASCII text file divided into sections. Each section can have a number of variables with associated values separated by an equal sign. The variable names are case insensitive. Comment lines are allowed and are prefixed by a sharp (#) sign. The variable names are case insensitive. If the variable is a file name, the case of the name is retained: this can be significant on operating systems that have a case-sensitive file system. An example is given below:

```
[State]
SnowpWE     =10
SnowpDens=11
SnowpDepth=12
IntercStore=1.2
SMD=23
GroundWStore=23
```

```
ResStorage  =12
ResRelease  =23
ChanStorage=0.2
FloodpStorage=1.2
Qsimulated=   2.513153
Qupdated=   6.814422

# always 50 flows in transit
[FlowsInTransit]
flow_01 = 12
flow_02 = 45
flow_03 = 0
flow_04 = 0
…
flow_50 = 0
```

The following sections are required in this file:

```
[State]
[FlowsInTransit]
```

Note that in order to support the transition from the previous version of this model (hourly data), state files containing the previous `[HourlyFlowsInTransit]` are supported, but the output state file will use the new `[FlowsInTransit]` section header.

The following tables list the variables that must be present in each section:

[State]			
Variable	**Description**	**Type**	**Required**
`SnowpWE`		real	YES
`SnowpDens`		real	YES
`SnowpDepth`		real	YES
`IntercStore`		real	YES
`SMD`	Soil moisture deficit	real	YES
`GroundWStore`		real	YES
`ResStorage`		real	NO (needed only if a reservoir exists)
`ResRelease`		real	NO (needed only if a reservoir exists)
`ChanStorage`		real	YES
`FloodpStorage`		real	YES
`Qsimulated`	Last value of the simulated Q of the previous hindcast period	real	YES
`Qupdated`	Last value of the simulated Q of the previous forecast period	real	YES

[FlowsInTransit]			
Variable	**Description**	**Type**	**Required**
`flow_01`	flow point 1	real	YES
`flow_02`	flow point 2	real	YES
`flow_03`	flow point 3	real	YES
…			
`flow_50`	flow point 50	real	YES

Note that the zero padding for flows below 10 is ***required!***

3.2 MCRM Output

The MCRM module produces the following output and files:

- Module log file
- Return Code of module executable
- Output time-series files (in csv format)
- Module state at the end of the hindcast period

3.2.1 Time series files

Output time-series are stored in two separate csv files, one for the forecast period and one for the hindcast period. For the hindcast period, the file contains: simulated discharge, updated discharge, soil moisture deficit, groundwater, reservoir storage and reservoir release. For the forecast period, the file contains: simulated discharge, updated discharge, soil moisture deficit, groundwater, Snow Water Equivalent, Snow density, reservoir storage and reservoir release. Reservoir storage and Reservoir release are only written if a reservoir is present.

The number of records in the hindcast file should match the number of time-steps within the backhours period while the number of records in the forecast file should match the number of time-steps contained within the forchours period. Columns are separated by a comma (,). Comment lines are allowed and are prefixed by a sharp (#) sign. Example files for the forecast and the hindcast period are given below:

```
Outforecast.csv:

#Q.sim,Q.update,SMD,Gwater,SnowWE, snowDen, ResStor,ResRelease
0.2,0.2,23.0,200,3000,1

Outhindcast.csv:

# Q.sim,Q.update,SMD,Gwater,snowWE,snowDen,ResStorage,ResRelease
0.2,0.2,23.0,200,3000,1
```

3.2.2 Model state file

This file contains the state of the model *after* the hindcast period has been executed (from start to backhours). The format of this file is identical to the initial state format (see section 3.1.7).

3.2.3 Return code

See next section.

3.2.4 Log Message file

During program execution the module writes messages to a plain ASCII log file with name and location as specified in the `data.properties` file. If a fatal error occurs that cannot be logged or if the log file cannot be opened the module stops execution and returns a non-zero (1) exit code. If a log file cannot be made at the specified location a log file will be created in the directory where the MCRM executable resides and the module will return a non-zero exit code. The MCRM log file resembles the PI diagnostics file format. The warning levels in the diagnostics file will be interpreted by the general adapter according to the following table:

Prefix	Description	Example
debug	Debug information	DEBUG: Calling error model
info	information, all is well	INFO: MCRM normal termination
warning	warning information	WARNING: MCRM unresolved symbol
error	critical problems	ERROR: Cound not file variable ShowDepth in section parameters
fatal	fatal error, complete module crash	FATAL: Unable to open parameter file

Example of a log file for a successful run of MCRM:

```
INFO: Midlands MCRM V 1.00.06 of 21 November, 2003
INFO: File names:..\config\catchment.properties          ..\config\data.properties
 INFO: catchment structure read in progress
 INFO: Reading catchment structure file completed
 INFO: Catchment parameters reading in progress
 INFO: Reading catchment parameters completed
 INFO: Start reading input Data
 INFO: Open and read the catchment hindcast time series input file
 INFO: Done reading the hindcast time series input file
 INFO: Open and read the catchment forecast time series input file
 INFO: Done reading the forecast time series input file
 INFO: Open and read the catchment state file
 INFO: Done reading the catchment state file
 INFO: End of reading catchment and input and state Data
 DEBUG: starting HAFREN
 DEBUG: call SnowPackModel  -2.000000
 DEBUG: call SoilStorModel
 DEBUG: call GwatStorModel
 DEBUG: call LagCatchRoff
 DEBUG: call CatchFlowHyd
 DEBUG: call CatchErrorForcModel
 WARN: no error updating, missing Qobs in hindcast (HAFREN)
 DEBUG: Starting to write output...
 INFO: The Catchment output data is being written to file
 INFO: Normal Completion of Program MCRM
```

Example of a failed run:

```
 INFO: Midlands MCRM V 1.00.06 of 21 November, 2003
 INFO: File names:..\config\catchment.properties          ..\config\data.properties
 INFO: catchment structure read in progress
 INFO: Reading catchment structure file completed
 INFO: Catchment parameters reading in progress
 ERROR: (read_real) could not read MaxTempSnowfall in section Parameters
 ERROR: Invalid MaxTempSnowfall
 INFO: Reading catchment parameters completed
 FATAL: ERRORS FOUND in reading catchment parameters: Exiting
```

3.2.5 Array lengths in the migrated module

The following lengths of model run are currently supported. When run at a 15 minute interval this maintains a support level of 960 hours hindcast and 480 forecast. For hourly data, the supported periods are now 4 times as long.

Period	Previous version	Current version
hindcast	960 hours	3840 time-steps (=960 hours at 15 min)
forecast	480 hours	1920 time-steps (=480 hours at 15 min)

References

Bailey, R.A. and Dobson, C. 1981. Forecasting for floods in the River Severn catchment. *J. Inst. Water Eng. and Scientists*, **35**(2), 168-178.

Delft Hydraulics and Tessella 2005. National Flood Forecasting System. MCRM Module and Adapter User and System Documentation. Version 1.1, Report prepared for the Environment Agency.

Harding, R.J. and Moore, R.J. 1988. Assessment of snowmelt models for use in the Severn Trent flood forecasting system. Contract report to Severn Trent Water Authority, December 1988, Institute of Hydrology, 41pp.

Moore, R.J and Bell, V.A. 2001. Comparison of rainfall-runoff models for flood forecasting. Part 1: Literature review of models. Environment Agency R&D Technical Report W241, Research Contractor: Institute of Hydrology, September 2001, Environment Agency, 94pp.

Moore, R.J., Harding, R.J., Austin, R.M., Bell, V.A. and Lewis, D.R. 1996. Development of improved methods for snowmelt forecasting. R&D Note 402, Research Contractor: Institute of Hydrology, National Rivers Authority, 192pp.

Wallingford Water 1994. A flood forecasting and warning system for the River Soar: Stage 2 Report. Contract report to the National Rivers Authority Severn Trent Region, Version 2.0, January 1995, 99 pp plus Appendices, Wallingford Water, Wallingford, UK (available from Centre for Ecology & Hydrology, Wallingford).