AGU PUBLICATIONS

Water Resources Research

RESEARCH ARTICLE
10.1002/2016WR019092

Key Points:
- Improved snow simulations by snow depth assimilation with particle filter
- Possible to estimate gauge undercatch for snowfall using the particle filter
- Implementation of particle filter efficient and compatible with complex nonlinear models

Correspondence to:
J. Magnusson,
jmg@nve.no

Citation:
Magnusson, J., A. Winstral, A. S. Stordal, R. Essery, and T. Jonas (2017), Improving physically based snow simulations by assimilating snow depths using the particle filter, *Water Resour. Res.*, *53*, doi:10.1002/2016WR019092.

Received 19 APR 2016
Accepted 20 DEC 2016
Accepted article online 29 DEC 2016

Improving physically based snow simulations by assimilating snow depths using the particle filter

Jan Magnusson[1,2], Adam Winstral[2], Andreas S. Stordal[3], Richard Essery[4], and Tobias Jonas[2]

[1]Norwegian Water Resources and Energy Directorate, Oslo, Norway, [2]WSL Institute for Snow and Avalanche Research SLF, Davos, Switzerland, [3]IRIS, Stavanger, Norway, [4]School of GeoSciences, University of Edinburgh, Edinburgh, UK

Abstract Data assimilation can help to ensure that model results remain close to observations despite potential errors in the model, parameters, and inputs. In this study, we test whether assimilation of snow depth observations using the particle filter, a generic data assimilation method, improves the results of a multilayer energy-balance snow model, and compare the results against a direct insertion method. At the field site Col de Porte in France, the particle filter reduces errors in SWE, snowpack runoff, and soil temperature when forcing the model with coarse resolution reanalysis data, which is a typical input scenario for operational simulations. For those variables, the model performance after assimilation of snow depths is similar to model performance when forcing with high-quality, locally observed input data. Using the particle filter, we could also estimate a snowfall correction factor accurately at Col de Porte. The assimilation of snow depths also improves forecasts with lead-times of, at least, 7 days. At further 40 sites in Switzerland, the assimilation of snow depths in a model forced with numerical weather prediction data reduces the root-mean-squared-error for SWE by 64% compared to the model without assimilation. The direct insertion method shows similar performance as the particle filter, but is likely to produce inconsistencies between modeled variables. The particle filter, on the other hand, avoids such limitations without loss of performance. The methods proposed in this study efficiently reduces errors in snow simulations, seems applicable for different climatic and geographic regions, and are easy to deploy.

1. Introduction

Physically based snow models play an important role in applications such as flood and avalanche forecasting [*Brun et al.*, 1992; *Zhao et al.*, 2009]. Data assimilation methods can efficiently reduce errors in snow simulations arising from uncertainties in, for example, the forcing data [e.g., *Leisenring and Moradkhani*, 2011; *Magnusson et al.*, 2014]. However, though data assimilation has been incorporated in many simpler snow model applications [e.g., *Brown et al.*, 2003; *Slater and Clark*, 2006], data assimilation in multilayer snow models is more challenging [e.g., *Andreadis and Lettenmaier*, 2006; *Durand et al.*, 2009] and still has potential for improvements. Reducing the uncertainty in predictions given by physically based snow models would be valuable since the simpler models have limited use for flood forecasting in situations that deviate from typical weather patterns or when conditions push the limits of their assumptions [e.g., *Kumar et al.*, 2013; *Rössler et al.*, 2014]. Additionally, avalanche forecasting typically requires complex modeling platforms in order to detail the snowpack layering that controls snow stability; simpler models often cannot be relied upon for this [*Bartelt and Lehning*, 2002].

In this study, we use a particle filter, which is a Bayesian recursive estimation technique for state and parameter updating [*Arulampalam et al.*, 2002]. This method approximates the probability distributions using a finite set of samples, so-called particles. The method is robust to nonlinearity and nonnormal distributions. Furthermore, the particle filter does not alter the states of the individual particles. Each particle maintains its modeled states with only their associated weights updated based on the assimilated data. The weights then determine best estimates of the current snow state. This property is convenient for multilayer snow models to ensure that the simulated snowpack is consistent with the observed data.

Recently, in data assimilation experiments using synthetic observations, *Charrois et al.* [2015] tested the particle filter algorithm for the Crocus model, which is a sophisticated snow model used for avalanche forecasting and hydrologic modeling [*Vionnet et al.*, 2012]. The assimilation of visible and near infrared reflectance

data reduced the uncertainty in the simulations of snow depth and snow water equivalent (SWE). An even greater reduction in the errors of the simulations was achieved by updating the model using snow depth observations. *Dechant and Moradkhani* [2011] evaluated the particle filter for updating the states of the simple model SNOW-17 using remotely sensed microwave radiance data. The particle filter algorithm improved simulations of SWE as well as streamflow forecasts. Using the same model, *Leisenring and Moradkhani* [2011] successfully applied the particle filter for directly assimilating snow water equivalents observed at one station in the United States. *Thirel et al*. [2013] also used the particle filter algorithm to update snow simulations of the LISFLOOD model by assimilating satellite observations of snow covered area. Their experiments also resulted in improved streamflow predictions. Finally, *Margulis et al*. [2015] showed that a particle smoother improved SWE estimates when assimilating observations of fractional snow covered area. This method was recently used for generating a high-resolution SWE reanalysis for the Sierra Nevada mountain range using Landsat data [*Margulis et al*., 2016]. Thus, the particle filter algorithm seems reliable for updating snow models, but has yet to be used for assimilating real observations of snow depths, which is one of the easiest and cheapest variables of the snowpack to measure at single points.

In this study, we examine the applicability of the particle filter for a multilayer energy-balance snow model using snow depth observations. Our main goal is to test whether the assimilation of snow depths improves estimates of variables such as SWE and snowpack runoff. Typically, such snow depth measurements are readily available and cost efficient whereas other more hydrologically relevant variables, for example, SWE and snowpack runoff, are much more difficult and expensive to obtain. A key difference between the current study and *Charrois et al*. [2015] is that we use actual snow depth observations whereas the latter study used synthetic measurements. We test the particle filter in detail at one field site with long-term data available for snow research. Furthermore, we also evaluate the method at field sites with realistic conditions for applications where only weather forecasting data are available for driving the model. Finally, as a benchmark method, we compare the performance of the particle filter against direct insertion of observed snow depths. This method has been used in several previous studies for adjusting state variables of snow models to match observations [e.g., *Bartelt and Lehning*, 2002; *McGuire et al*., 2006; *Fletcher et al*., 2012].

2. Study Site and Data

We first test the particle filter method at the Col de Porte field site in France, using input data sets of different quality. In a second experiment, we evaluate the data assimilation scheme at 40 stations in Switzerland. All stations are in operational use for avalanche and flood forecasting.

2.1. Data From Col De Porte

We use a published data set for the Col de Porte field site (45.30°N, 5.77°E) that is located at an elevation of 1325 m in France. The data set is publically available and described in detail by *Morin et al*. [2012]. This field site has long-term observations of meteorological and validation variables necessary for detailed evaluations of snow models [e.g., *Essery et al*., 2013; *Magnusson et al*., 2015; *Wever et al*., 2014]. In this study, we perform snow simulations for the period from 1 October 1994 to 30 June 2010. Typically, the snow cover lasts from roughly December to April at Col de Porte and mid-winter melt events are common.

2.1.1. Model Input Data—In Situ Meteorological Data

The following meteorological variables required for input to snow models have been recorded at a weather station directly at the site with an hourly resolution: air temperature, relative humidity, wind speed, precipitation using a heated gauge, incoming longwave and shortwave radiation. In the available data set, precipitation is divided into rain and undercatch-corrected snowfall (see *Morin et al*. [2012] for details). The precipitation partitioning has been performed manually using auxiliary information, foremost air temperature, snow depth and relative humidity. However, in many applications, the precipitation phase is determined from air temperature alone, as manual corrections such as these require great effort and comprehensive auxiliary data. Therefore, to mimic typical situations, we compute the precipitation phase using air temperature, and compute total precipitation from the sum of rainfall and snowfall. Additionally, for the particle filter, we can also generate an ensemble of inputs to the model that preserves the physical consistency between precipitation phase and air temperature with this approach.

We determined the fraction of snowfall, sf (unitless), from air temperature, T (°C), and a threshold temperature, T_{base} (°C), below which precipitation falls mainly as snow:

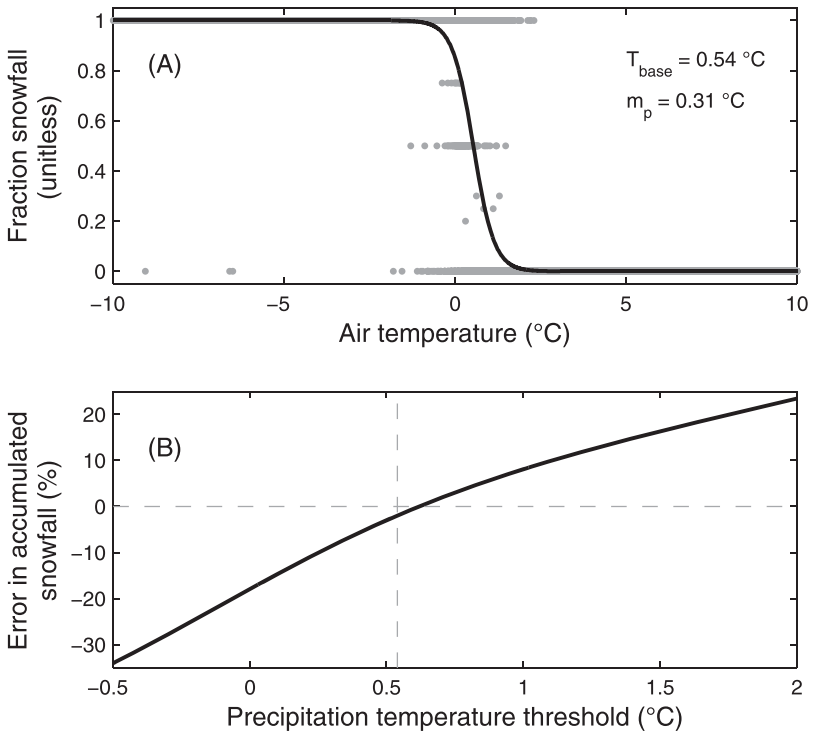

Figure 1. (top) The snowfall fraction computed using the rainfall and snowfall records in the publically available data set (grey dots) and the curve fitted to those data (black line). (bottom) The sensitivity of the accumulated snowfall over the complete study period as a function of precipitation threshold temperature (black line). The calibrated temperature threshold, represented by the vertical grey dashed line in the bottom plot, gives a slight underestimation of total snowfall.

$$sf = \frac{1}{1+\exp\left(T_p\right)} \tag{1}$$

where $T_p = (T - T_{base})/m_p$. The parameter m_p (°C) determines the temperature range for mixed precipitation. See *Kavetski and Kuczera* [2007] for further details.

At Col de Porte, the ratio of snowfall to total precipitation shows that mixed precipitation occurs in a temperature range from approximately −1.8°C to 2.3°C (grey dots in Figure 1a). We estimated the parameters of equation (1) by minimizing the sum of squared errors between simulated and observed snowfall ratio (see black line in Figure 1a). We also assessed the sensitivity of the estimated total snowfall using equation (1) by varying threshold temperature, T_{base}, and compared with the total snowfall amount given in the publically available data set (see Figure 1b). This analysis shows that at this site small differences in the threshold temperature can produce large biases in the total snowfall estimates.

2.1.2. Model Input Data—SAFRAN Reanalysis System

For Col de Porte, the same meteorological variables as recorded by the weather station are also available with hourly resolution from an analysis system called SAFRAN [*Durand et al*., 1993]. This system pragmatically mixes in situ meteorological information, radiosondes and output from large-scale weather prediction models to provide an optimal analysis within meteorologically homogeneous areas (termed massifs) of the French Alps. Each massif has an average surface area of approximately 1000 km^2, and spans an elevation range of 300 m. The publically available data set contains SAFRAN data, which we use in this study, interpolated to the altitude of the Col de Porte field site. However, note that the SAFRAN analysis used in the data set of *Morin et al*. [2012] does not use observations from Col de Porte itself.

2.1.3. Data for Assimilation and Model Evaluation

In this study, we used the following variables from the Col de Porte data set for either assimilation or evaluation of the model: snow depth, SWE, snow lysimeter runoff, and soil temperature. The field site is equipped with two lysimeters. In this study, we use data from the lysimeter with the largest surface collection area, which equals 5 m^2. The lysimeters do not include a soil column, and therefore measure the runoff from the

snowpack directly without time delay. Snow depth measurements were automated and collected at an hourly time step. SWE was measured manually on a weekly basis in snow pits. Note that the SWE recordings were made independently from the automatic snow depth recordings. Additionally, we made use of soil temperature measured at 10 cm depth below the ground surface. For details about measurement methods and sensor specifications, please see *Morin et al*. [2012].

2.2. Data From Switzerland

We use data from 40 stations situated in Switzerland spanning an elevation range from 1195 to 2690 m.a.s.l and covering the period from 10 September 2013 to 31 July 2015. At these locations, field observers perform manual snow observations, such as snow depth (daily), SWE (bimonthly), and stability assessments (bimonthly) of the snowpack. Flat terrain and small exposure to strong winds characterize these sites, so that the snow observations should reflect the average conditions for a larger area. The avalanche warning service and the operational flood forecasting team in Switzerland regularly use data from these stations for their assessments. Thus, filling the gaps in between the occasional manual observations, as well as inferring other variables from those measurements using models in combination with data assimilation is of great practical importance.

2.2.1. Model Input Data—COSMO Weather Forecasting Model

Most of these sites in Switzerland lack meteorological measurements and, instead, we drive the model with data from the numerical weather forecasting model COSMO. COSMO, used operationally by MeteoSwiss (www.meteoswiss.ch), provides the full set of input variables required by energy-balance snow models on a horizontal grid with resolution of approximately 2 km. Though COSMO assimilates station data, none of the stations included in this study are part of the assimilated data set. We linearly interpolated the results from the COSMO grid to the location of the field sites. For air and dew point temperature, we corrected the COSMO results to the elevation of the field locations using a fixed linear lapse rate (0.006°C/m for both air and dew point temperature). Finally, using the air and dew point temperature we computed elevation corrected relative humidity.

2.2.2. Data for Assimilation and Model Evaluation

Every morning during winter, a field observer measures snow depth on a fixed graded rod at the field sites. Additionally, SWE measurements are made in a pit close to the snow rod approximately every second week. We use the snow depth measurement at the rod for assimilation, and the independent pit SWE observations for evaluation.

3. Description of the Snow Model

In this study, we use an energy-balance snow model provided in a multimodel framework [*Essery et al*., 2013]. This framework called JIM (JULES investigation model), solves the mass and energy exchanges for three individual snow layers, and is thus similar to many snow models applied in past research efforts [e.g., *Shrestha et al*., 2010; *Zanotti et al*., 2004]. The surface heat balance equation is solved analytically and vertical temperature profiles in the snowpack and the soil are solved by the Crank-Nicolson method. The model has five soil layers extending to a depth of 2 m and uses a variable number of snow layers, up to a maximum of three for snow more than 0.5 m deep. The total depth of the snowpack is derived from the simulated mass and density of the individual snow layers, and the modeled SWE is given by the sum of the snow mass for all layers. In this study, we run the model using an hourly time step, whereas we assimilate daily snow depths as outlined in section 4.

Due to the multimodel formulation, JIM gives the user different choices for simulating internal snowpack processes such as the settling of snow due to compaction and the time evolution of the snowpack surface albedo. For seven processes, the user can choose among three different methods, indicated by a number from 0 to 2. The list below shows those processes, and includes information about which method we selected in this study.

1. *Snow compaction*: we use a physically based method described by *Anderson* [1976], option 0 in JIM, to simulate the increase in snow density for each of the snow layers due to metamorphosis and weight of overlying snow.
2. *New snow density*: we use the method presented by *Oleson et al*. [2004], given as option 1 in JIM, to simulate the density of newly fallen snow.

3. *Snow albedo*: we simulate the variations in reflectivity of snow using an empirical method following *Douville et al*. [1995], option 1 in JIM.
4. *Turbulent heat exchange*: we computed the turbulent exchange of heat and moisture between the snow and atmosphere using the Richardson parameterization following *Louis* [1979], option 1 in JIM.
5. *Snow cover fraction*: the snow cover fraction follows a linear relationship with snow depth between 0 and 10 cm, with complete snow coverage for depths exceeding 10 cm. This method is denoted option 2 in JIM.
6. *Snow hydraulics*: we modeled the process for routing liquid water through the snowpack following the methods of *Boone and Etchevers* [2001], given as option 0 in JIM.
7. *Thermal conductivity of snow*: we model the thermal conductivity of the snowpack using the methods of *Douville et al*. [1995], given as option 1 in JIM.

While there are 1701 feasible combinations of the different model setups, we identified this particular model configuration using the simulation results for Col de Porte presented in *Magnusson et al*. [2015]. This model configuration shows the lowest normalized root-mean-squared-deviation for SWE and snow depth.

4. Data Assimilation Methods

4.1. Particle Filter

4.1.1. General Description

The particle filter is a commonly used data assimilation technique that can handle both nonlinear models and all types of probability distributions. Below we give a short description of the method, and follow the notation of *Arulampalam et al*. [2002]. The particle filter originates from the sequential Bayesian estimation problem. The first step in this estimation problem consists of the time-update using a state-space model:

$$x_k = f(x_{k-1}, \theta_{k-1}, u_{k-1}) + v_{k-1} \tag{2}$$

$$z_k = h(x_k) + n_k \tag{3}$$

where f is the state function, h is the measurement function, x is the state vector, z is the measurement vector, θ is the model parameters, u is the model inputs, v and n is the process and measurement noise, and k is the time index. In our case, the snow model running on an hourly time step is the state-space model. The states consist of variables such as snow density and temperature of each individual snow layer, and no observation operator is required since snow depth is a model output (see *Essery et al*. [2013] for further details about the model state variables). The aim of the sequential filtering problem is to find the posterior distribution $p(x_k|z_{1:k})$, which is the conditional distribution of the current state given all available observations. If this posterior distribution is available for the previous time step, we can compute the prior probability density for the current time step as:

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1}) p(x_{k-1}|z_{1:k-1}) dx_{k-1} \tag{4}$$

With Bayeś law we can update this prior density using new observations:

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k) p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})} \tag{5}$$

The following equation gives the normalization constant:

$$p(z_k|z_{1:k-1}) = \int p(z_k|x_k) p(x_k|z_{1:k-1}) dx_k \tag{6}$$

For most models, we are not able to solve this problem analytically. We instead approximate the posterior filter density $p(x_k|z_{1:k})$ using Monte Carlo samples:

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{N_s} \omega_k^i \delta(x_k - x_k^i) \tag{7}$$

where each sample, a so-called particle, with index i has a weight ω. Here N_s is the number of particles, and $\delta(\cdot)$ is the Dirac delta function. We compute the weights of each particle using the following recursive relationship:

$$\omega_k^i = \omega_{k-1}^i p\left(z_k | x_k^i\right) \tag{8}$$

where $p\left(z_k | x_k^i\right)$ is the likelihood function, which is the probability density function of the measurement error. To avoid filter degeneracy, we resample the particles if the effective sample size:

$$N_{eff} = \frac{1}{\sum_{i=1}^{N_s} \left(\omega_k^i\right)^2} \tag{9}$$

falls below a specified number of particles using residual resampling (see *Douc et al*. [2005] for details about the resampling procedure). We vary the total number of particles N_s in some experiments (see sections 5.2–5.5), and therefore perform the resampling step if the effective sample size falls below 80% of N_s. After the resampling step, we give all particles equal weight. In this study, snow depth observations are available once a day and we consequently perform the update step at the hour of the measurement. For further details about recursive Bayesian estimation and the particle filter in particular, see *Arulampalam et al*. [2002].

4.1.2. Likelihood Function

The uncertainties of snow depth measurements include both measurement errors and representativeness errors of the point observations. Typically, snow depth measurements have small errors (few centimeters), but their representativeness is often poor due to large spatial variability of the snow cover. For the experiments in this study, we define a likelihood function representing both types of errors as:

$$p\left(z_k | x_k^i\right) = N\left(z_k - x_k^i, \sigma\right) \tag{10}$$

$$\sigma = \max\left(0.10\ z_k, 5\right) \tag{11}$$

where N is the normal probability distribution of the residuals between observed, z_k, and simulated, x_k, snow depth in unit centimeters. We define the standard deviation, σ, of this probability distribution proportionally to the observed snow depth. For the plot scale (10 by 10 m), *Lopez-Moreno et al*. [2011] found a coefficient of variation for snow depth equal to approximately 0.10 for open fields, which we use as proportionality constant in this study. For snow depths below 50 cm, we use a fixed standard deviation of 5 cm. Note that the particle filter performance, as is the case in most data assimilation schemes, strongly depends on the defined uncertainties of the observations.

4.1.3. Particle Generation

We generate particles by forcing the snow model with an ensemble (particles) of input data subject to stochastic noise, either additive or multiplicative, applied each hour. See the description below and Table 1 for details about how we generated the particles.

Following *Evensen* [2003], the time evolution of the errors q_k for time k in the input data can be represented as:

$$q_k = \alpha q_{k-1} + \sqrt{1-\alpha^2} w_{k-1} \tag{12}$$

where w_k is white noise with mean equal to 0 and variance equal to 1, and α determines the time decorrelation. We draw the initial error, q_0, from the standard normal distribution. We can express α using a time decorrelation length τ with the following relationship:

$$\alpha = 1 - \frac{\Delta t}{\tau} \tag{13}$$

where Δt is the model time step. Finally, we can shift and scale the time series of correlated noise to produce the desired mean and variance. Furthermore, we can transform the normally distributed noise q_k as specified above to lognormally distributed noise l_k using the following relationship:

$$l_k = \exp\left(\mu + \sigma q_k\right) \tag{14}$$

where μ is the mean and σ is the standard deviation of the associated normal distribution.

Following the approach by *Charrois et al*. [2015], we define the error statistics on the forcing data by comparing the SAFRAN results with the observations from the meteorological station situated at Col de Porte.

Table 1. Error Statistics for Generating Noise on Model Input Data for Air Temperature (Ta), Relative Humidity (RH), Shortwave Radiation (SW), Longwave Radiation (LW), Precipitation (P), and Wind Speed (Ua)[a]

Variable	Unit	Distribution	μ	σ	τ (h)	Lower Limit	Upper Limit
Ta	°C	Normal	0	0.9	4.8	NA	NA
RH	%	Normal	0	8.9	8.4	0	100
SW	W/m^2	Normal	0	min(SW,109.1)	3.0	0	NA
LW	W/m^2	Normal	0	20.8	4.7	0	NA
P	mm/h	Lognormal	−0.19	0.61	2.0	0	NA
Ua	m/s	Lognormal	−0.14	0.53	8.2	0.5	25

[a]For some variables, such as relative humidity, we constrain the perturbed data to a plausible range, here denoted as the lower and upper limit. We computed the error statistics, using data from Col de Porte covering the period from 1 August 1993 to 31 July 2011. The summer months, see white regions in Figure 4 in *Morin et al*., [2012], were excluded from this analysis.

For all variables except precipitation and wind speed, the residuals between the observations and the SAFRAN data are roughly symmetrically distributed. For those variables, we perturb the data using additive noise generated as described above. To estimate the spread in the residuals robustly, we use the median absolute deviation (*MAD*) instead of the standard deviation [*Leys et al*., 2013]:

$$MAD = b\ \ median(|y_i - \ \ median(y_i)|) \tag{15}$$

where y_i is the n observations. The constant b equals 1.4826 under the assumption of normally distributed data, disregarding the influence introduced by eventual outliers [*Leys et al*., 2013]. Note that for shortwave radiation, we only compute the error statistics during daytime. For precipitation and wind speed, the ratio between the in situ and SAFRAN data series approximately follows a lognormal distribution during times when variables are respectively greater than 1 mm and 1 m s^{-1}. From the logarithm of this ratio, we compute the parameter σ in equation (14) again using the *MAD* to avoid strong influence of outliers. Afterward, we define the parameter μ in equation (14) so that the mean of the lognormal probability distribution equals one. We perform the last step to avoid introducing any biases caused by the sampling procedure. Note that we, following *Charrois et al*. [2015], assume that the SAFRAN data reproduces the timing of events accurately but not the amounts. Finally, for all variables, we computed the lag-one autocorrelation, which equals α in equation (12), from the residuals between the in situ and SAFRAN data. In Table 1, we present this parameter as a time decorrelation length τ given by the relationship in equation (13).

In this study, we use the same methods and error statistics as presented above for the SAFRAN data for representing the uncertainty in the COSMO input data at the locations in Switzerland. This approach likely lowers the assimilation efficiency of the particle filter method at the Swiss sites.

Note that we do not add any perturbations to, for example, the state variables. For the results presented in sections 5.1–5.5 and 5.8, the whole spread of the particles is due to the perturbations of the forcing variables alone.

4.2. State and Parameter Estimation: Assessing Snowfall Bias

Snow models are typically more sensitive to biases than random errors in the input data [*Raleigh et al*., 2015]. The perturbations on the forcing data introduced above represents random but not systematic errors in the inputs. Snow models, as are all hydrological models, are most sensitive to biases in the precipitation inputs. Using the particle filter, we perform a combined state and parameter estimation [*Kantas et al*., 2015] to assess potential precipitation biases at the Col de Porte site (see section 5.6). We also use this method for reducing the influence of any biases in solid precipitation in the results presented in sections 5.7 and 5.9.

For snow models, the simulated peak in SWE is obviously very sensitive to biases in snowfall rates. With the particle filter, it is possible to estimate such biases by augmenting the state vector with a parameter, in this case a snowfall correction factor, which we update along with the state variables. For snowfall, we assume a multiplicative bias, with a uniform prior distribution having a range from −75 to 300% following *Raleigh et al*. [2015]. To avoid degeneracy of the algorithm, we introduce artificial dynamics at each time step on the snowfall correction factor, *sfc* (unitless), for each particle by adding normally distributed noise with mean equal 0 and variance equal to 1:

$$sfc_k = sfc_{k-1} + \varepsilon\, N(0,1) \tag{16}$$

where ε is a scaling factor requiring tuning [*Doucet et al*., 2001]. In this study, we chose the scaling factor to equal 0.005 by manual calibration until we achieve satisfactory performance of the filter.

At Col de Porte, we can test the combined state and parameter estimation method in detail since the data record is very long. Additionally, as validation, we can estimate the bias in the snowfall rates given by the SAFRAN data using the in situ observations. With this basic version of the particle filter algorithm, we could not estimate the bias in any of the other forcing variables besides precipitation at Col de Porte. Such failures of the algorithm may be due to compensating mechanisms within the snow model. For example, the model may give similar results if a bias in longwave radiation, either positive or negative, compensates for a systematic error in solar radiation of the opposite sign. Thus, the filter cannot detect the correct answer, which is the case where both forcing variables are unbiased, from the cases where the errors in the input data cancel each other out. With a more sophisticated version of the particle filter algorithm, which comes at higher computational cost, it might be possible to estimate the bias in several of the forcing variables at once [*Kantas et al*., 2015]. For estimating systematic errors in more than one forcing variable, it might also be necessary to assimilate more variables (i.e., snow surface temperature) than only snow depth.

In addition to the ensemble spread caused by the perturbations of the forcing variables, the noise on the snowfall correction factor introduces additional spread in the particles. This applies to the results presented in sections 5.6, 5.7, and 5.9.

4.3. Direct Insertion

As a benchmark, we compare the results from the particle filter with those from direct insertion of the observed snow depths into the model. Simulated and measured snow depths are compared at each insertion time to determine the necessary increments. Positive depth increments are added to the snowpack with the temperature and density of the existing surface snow layer. Negative mass increments are calculated according to the fraction of a layer's depth that is removed. Snow is simply removed by negative increments and not added to runoff. After increments have been applied, snow is redistributed between layers according to the model's rules for assigning layer depths. Since the selected snow cover fraction model option reduces the fraction of snow cover when snow depths are less than 10 cm, direct insertion of observed snow depths only takes place when the observation exceeds 10 cm.

5. Results and Discussion

5.1. Time Series Examples From Col de Porte

From sections 5.1 to 5.5, we compare the performance of the data assimilation methods under two contrasting input data situations. The in situ data set is of very high quality, typical of well-equipped research sites, whereas the SAFRAN data set better represents the situation for practical applications. During most winters, the deterministic simulation using SAFRAN input data gives poorer results than the simulations driven by the meteorological data measured at the field site (see example in Figure 2). For evaluations of the complete study period, see sections 5.2–5.5. In the displayed case, the SAFRAN simulations underestimate the observed snow depth and SWE (Figures 2a and 2b). This underestimation produces a shorter spring snowmelt period (Figure 2c). Due to the early disappearance of snow in the SAFRAN simulation, the modeled soil temperatures increase too fast in spring. The simulation using the in situ data, on the other hand, agrees well with the measured snow depth and SWE (Figures 2a and 2b). For snowpack runoff, the in situ model run captures the runoff during April better than the SAFRAN simulation, but seems to underestimate snowmelt during the second half of March (Figure 2c). For soil temperature, the in situ simulation matches the observations well during the displayed period, but underestimates the measurements in early winter as well as in late April due to lagged snow disappearance (Figure 2d). This example illustrates how sensitive model outcomes are to the accuracy of meteorological forcings (see also studies such as *Raleigh et al*. [2015]).

After assimilation using the particle filter, the SAFRAN simulations closely follow the observed snow depths (Figure 3a). The simulated SWE also matches the observations better than the deterministic run (Figure 3b). For the spring snowmelt period, the particle filter seems to improve the runoff simulations, however, runoff is still underestimated in some situations (Figure 3c). For soil temperature, the model results match the

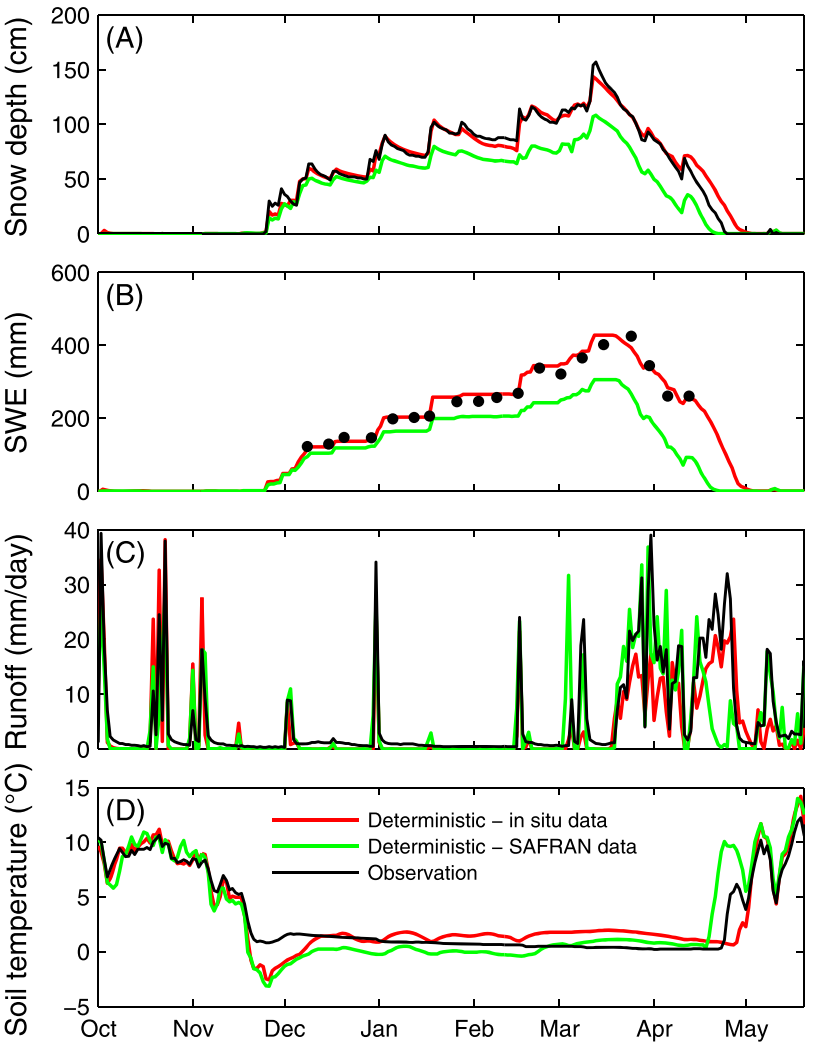

Figure 2. Typically, the deterministic simulation using in situ input data shows a better match with the observations, in particular for snow depth and SWE, than the deterministic simulation using SAFRAN data. The graph shows the results for the winter 2005/2006 at Col de Porte.

observations better after snow disappearance in the filter simulation than in the deterministic case (Figure 3d). Thus, this example shows that the assimilation of snow depths using the particle filter can improve the quality of several modeled snowpack variables apart from snow depth itself and even for variables of the soil column influenced by the snow cover.

5.2. Snow Depth Simulations at Col de Porte

Over the complete study period, the deterministic simulation of snow depth using SAFRAN input data shows 49% higher RMSE and 5% lower squared correlation coefficient than the same simulation using in situ input (Figure 4). The particle filter efficiently reduces the errors for both input data cases, even for a small number of particles. The performance of the filter seems to stabilize when using more than 100 particles. For the simulations using 2000 particles, the RMSE for snow depth decreases by 70% for the in situ simulations and 74% for the SAFRAN simulations compared to the respective deterministic runs. For the particle filter, we computed the RMSE using the best estimate of snow depth, which we obtained as the weighted average of the ensemble members using the particle weights. The correlation increases almost to one for both sets of input data. Our results show that the model results, after applying the particle filter method, tracks the snow depth development over the season closely. The direct insertion method replaces modeled snow depth with measured values and thus reproduces the observations perfectly (expect for snow depths lower than 10 cm; see description in section 4.3).

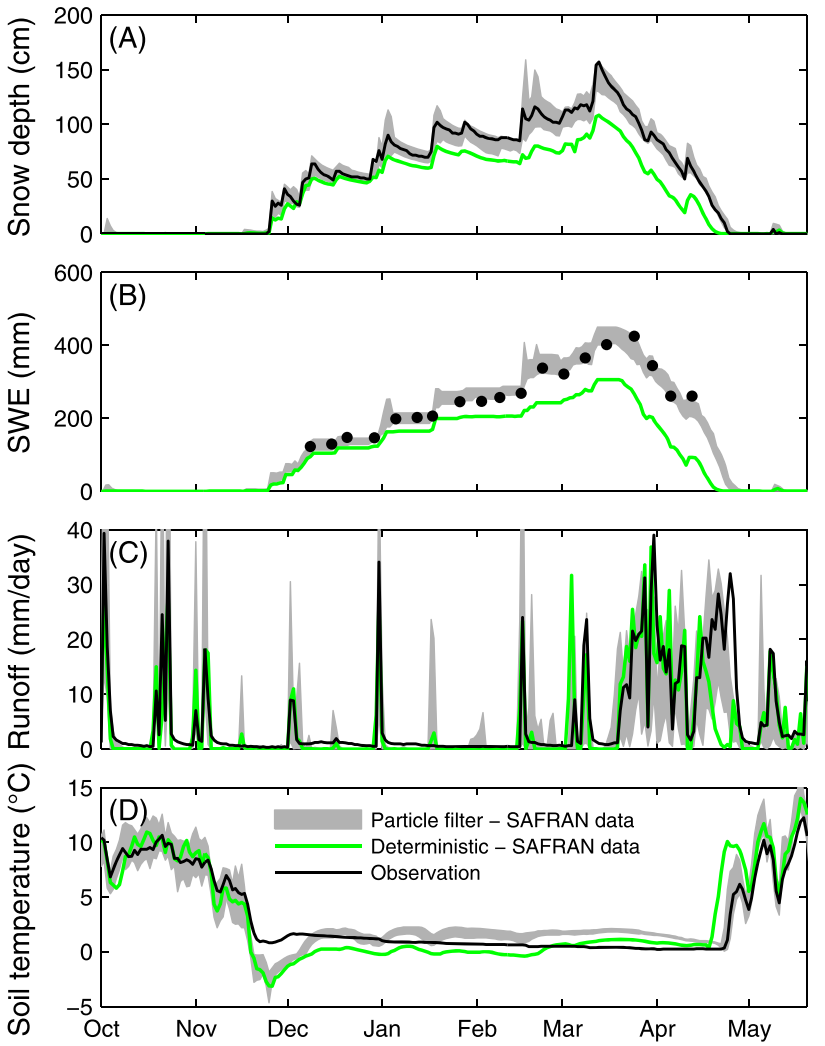

Figure 3. The particle filter with 100 ensemble members (grey area shows the range covered by the particles) improves the results from the deterministic run using SAFRAN input data. The graph shows the results for the winter 2005/2006 at Col de Porte.

5.3. Snow Water Equivalent Simulations at Col de Porte

For SWE, the deterministic simulations show approximately 107% higher RMSE and 19% lower squared correlation coefficient when using SAFRAN data as model input instead of in situ data (Figure 5). Thus, the lower quality of the data obtained by the SAFRAN system than observed by the weather station seems to have a larger impact on SWE than snow depth simulations. The model runs using SAFRAN data typically underestimate SWE (see example in Figure 2), and at the same time seems to underestimate snow density, which leads to better results for snow depth than SWE. For the SAFRAN data, the particle filter greatly reduces the errors. Similar as for snow depth, the performance of the filter seems to stabilize when using more than 100 particles. For 2000 particles, the filter reduces RMSE by 47% for the simulations using the SAFRAN data set, whereas the filter degrades the performance by approximately 10% for the model runs using in situ input data. The correlation, on the other hand, improves for both input data sets when using the particle filter. After applying the filter algorithm, the simulation results show almost equal performance between the two forcing data sets.

For the SAFRAN data set, direct insertion reduces the RMSE for SWE from 59 to 51 mm compared to the particle filter (2000 particles), but shows the same squared correlation coefficient. The small difference between the performance metrics for the particle filter and direct insertion shows that both methods produces almost the same best estimates for SWE.

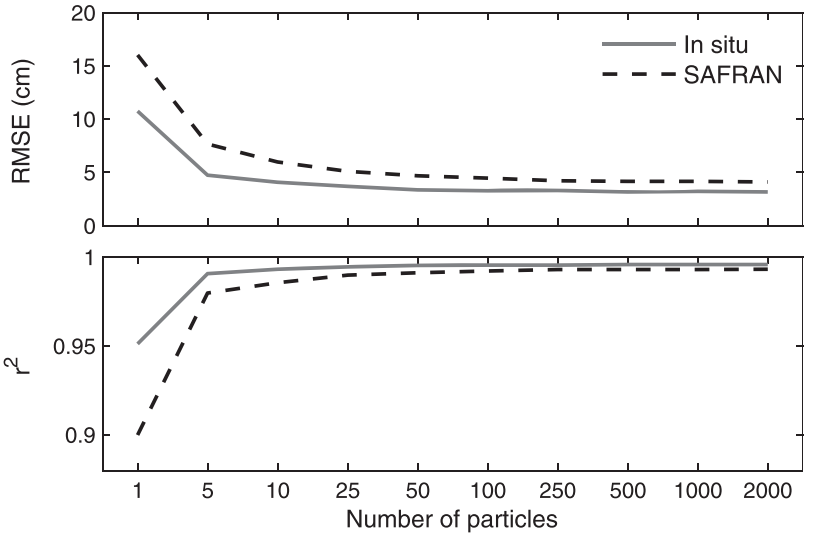

Figure 4. Performance of the particle filter at Col de Porte for snow depth and varying number of particles. (top) The root-mean-squared-error, and (bottom) the squared correlation coefficient. The deterministic run, generated without perturbed input data, is denoted as one particle. This evaluation uses 3886 snow depth measurements observed at mid-night typically from October until June each year over the period from 1 October 1994 to 30 June 2010. The particle filter reduces the errors efficiently compared to the deterministic simulation, which do not include any data assimilation.

5.4. Snowpack Runoff Simulations at Col de Porte

For snowpack runoff, the deterministic simulations show similar patterns in the results as for snow depth and SWE (Figure 6). The deterministic run using SAFRAN data shows 25% higher RMSE, 20% lower squared correlation coefficient, and 50% higher volumetric error than the simulations using in situ data. We compute the volumetric error from the ratio between simulated and observed total runoff for the whole study period. For the SAFRAN data set, the number of particles influences the results similarly as for snow depth and SWE; the performance of the simulations steadily increases with number of particles. On the other hand, when forcing the model with in situ data, the filter degrades the results slightly, in particular when using fewer than 50 particles. However, for both input data cases, the volumetric error decreases for the particle filter simulations, in particular for the SAFRAN runs. Thus, for daily runoff predictions, the filter gives

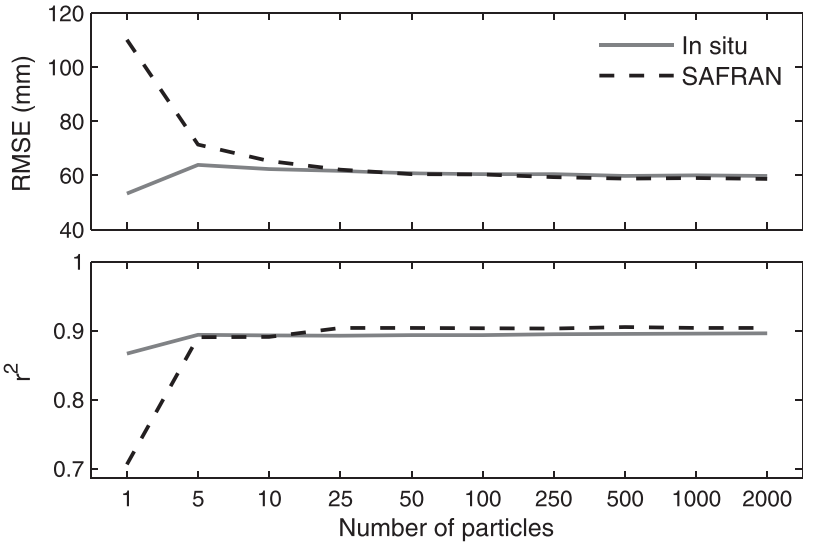

Figure 5. Performance of the particle filter at Col de Porte for SWE and varying number of particles. The deterministic run, generated without perturbed input data, is denoted as one particle. This evaluation uses 266 SWE measurements observed once a week typically from December until May each year over the period from 1 October 1994 to 30 June 2010. The particle filter substantially reduces the errors for the SAFRAN input data set, while a slight increase in RMSE was observed for the in situ input data.

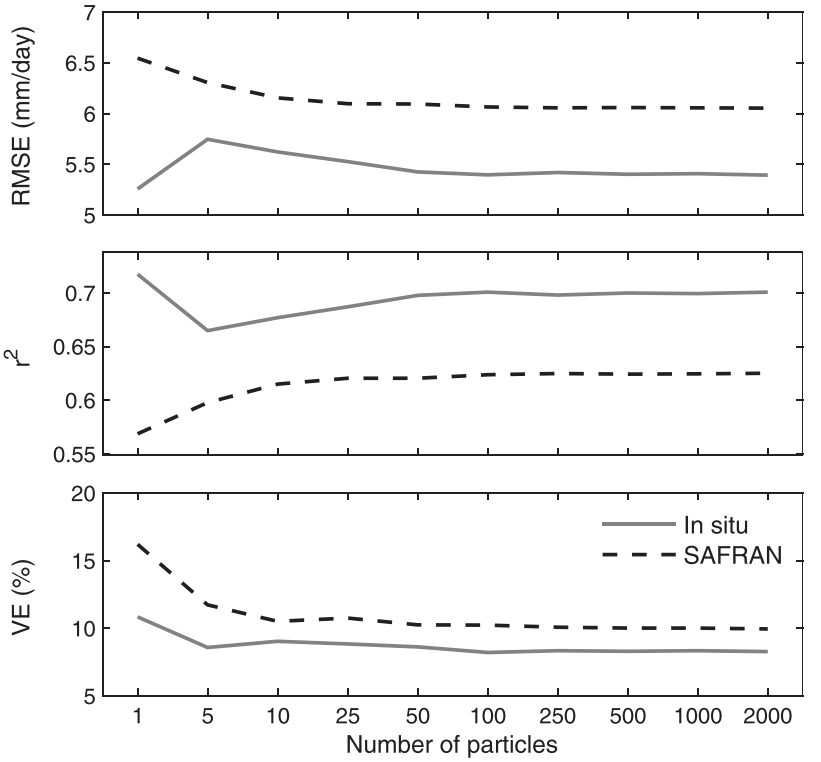

Figure 6. Performance of the particle filter at Col de Porte for snowpack runoff and varying number of particles. In addition to RMSE and squared correlation coefficient, we also judged the model performance using the volumetric error, which it the ratio between simulated and observed total runoff for the whole study period. This evaluation uses 3812 snowpack runoff observations, daily aggregates, typically observed from October until June each year over the period from 1 October 1994 to 30 June 2010. The deterministic run, generated without perturbed input data, is denoted as one particle.

ambiguous results depending on the quality of the input data. However, the reduction in volumetric error for both input data cases indicates that for longer averaging periods the simulated snowpack runoff becomes more accurate when using the data assimilation algorithm independent of the forcing data quality. Furthermore, rainfall events are common at Col de Porte even during winter. Approximately 30% of precipitation falls as rain during periods with snow cover. Thus, improvements in the daily melt computations may be masked by rainfall events, which the data assimilation methods cannot improve upon. Finally, also note that the lysimeter measurements at Col de Porte feature uncertainties and that measurement errors may influence the results [see *Magnusson et al.*, 2015, section 4.6].

For the SAFRAN forcing data, direct insertion shows higher RMSE (∼3%) and lower squared correlation coefficient (∼2%) than the particle filter method. The direct insertion results underestimates runoff with approximately 15%, whereas the corresponding value for the particle filter equals 10%. The volumetric errors are influenced by the setup of direct insertion in our study, which adjusts the snow state variables without corresponding changes in snowpack runoff (see section 5.9 for further discussion about the direct insertion implementation). The particle filter algorithm, on the other hand, maintains physical consistency between snow depth, SWE, and snowpack runoff.

5.5. Soil Temperature Simulations at Col de Porte

For soil temperature, the deterministic run using the SAFRAN input data shows almost the same overall performance as the simulation using the in situ data (Figure 7). The RMSE is only 2% higher and the squared correlation coefficient is just 3% lower for the SAFRAN simulation than the in situ run. The particle filter improves the results for both input data cases. The number of particles do not influence the performance of the filter strongly. For the SAFRAN data set, direct insertion shows slightly lower RSME (∼4%) and higher squared correlation coefficient (∼1%) than the particle filter with 2000 ensemble members. We find that the improvement in simulated soil temperature by applying the particle filter or direct insertion is less clear than for snow depth and SWE. The predicted soil temperatures are mainly influenced by the timing of

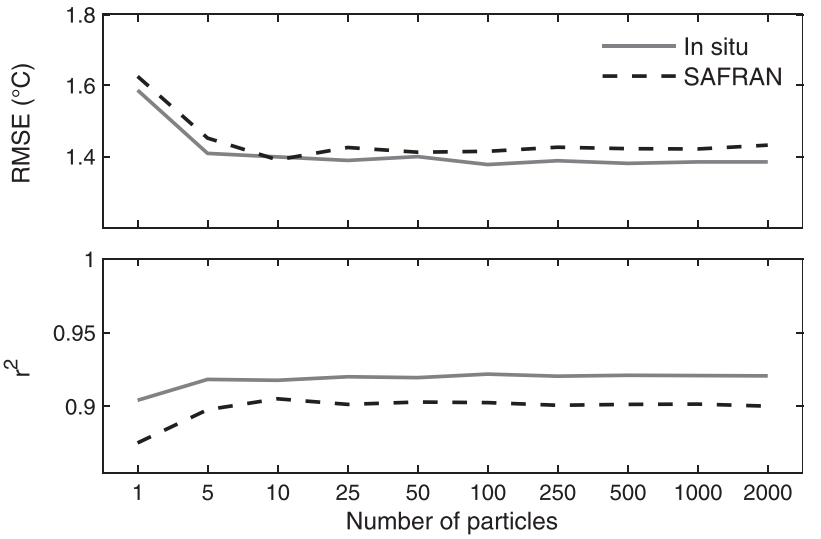

Figure 7. Performance of the particle filter at Col de Porte for soil temperature 10 cm below the ground surface and varying number of particles. The deterministic run, generated without perturbed input data, is denoted as one particle. This evaluation uses 3592 soil temperature observations, daily averages, typically observed from October until June each year over the period from 1 October 1994 to 30 June 2010. The filter reduces the errors for both input data sets.

seasonal snow coverage, which can differ between snowpack simulations (compare Figures 2d and 3d). This period is relatively short, and can explain the rather small difference between the simulation results presented in Figure 7. However, for modeling permafrost, the timing of snow disappearance is very important [*Luetschg et al.*, 2008]. Thus, for such applications, the particle filter or direct insertion can be a useful tool for improving the reliability of the results since the method ensures a better simulation of the snowpack.

5.6. Parameter Estimation at Col de Porte

The combined state and parameter updating using the particle filter provides an estimate of the snowfall correction factor (see grey shaded area in Figure 8). The filter algorithm narrows the parameter range toward the observed bias, estimated from the ratio of observed weather station to SAFRAN-predicted snowfall (see red dashed line). We also computed the same ratio for each winter separately (see black horizontal lines). In many winters, the estimated parameter value covers both the long-term average and the value given for each winter. However, in some years, the bias computed for the individual winters falls outside of this range. Three potential reasons that the filter may be unable to track the observed values during these winters are: (1) the observed bias is wrong, (2) the parameter estimation compensates for additional model

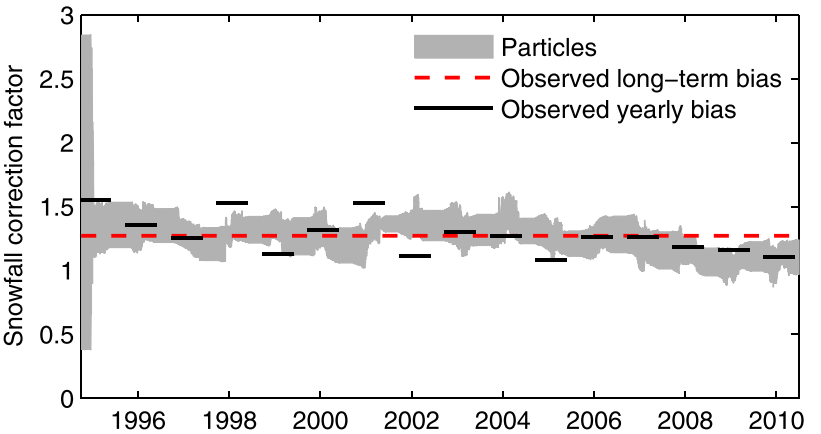

Figure 8. Evolution of the snowfall correction factor in the combined state-parameter estimation at Col de Porte using the particle filter with 2000 ensemble members. The grey shaded area represents the 95% confidence interval.

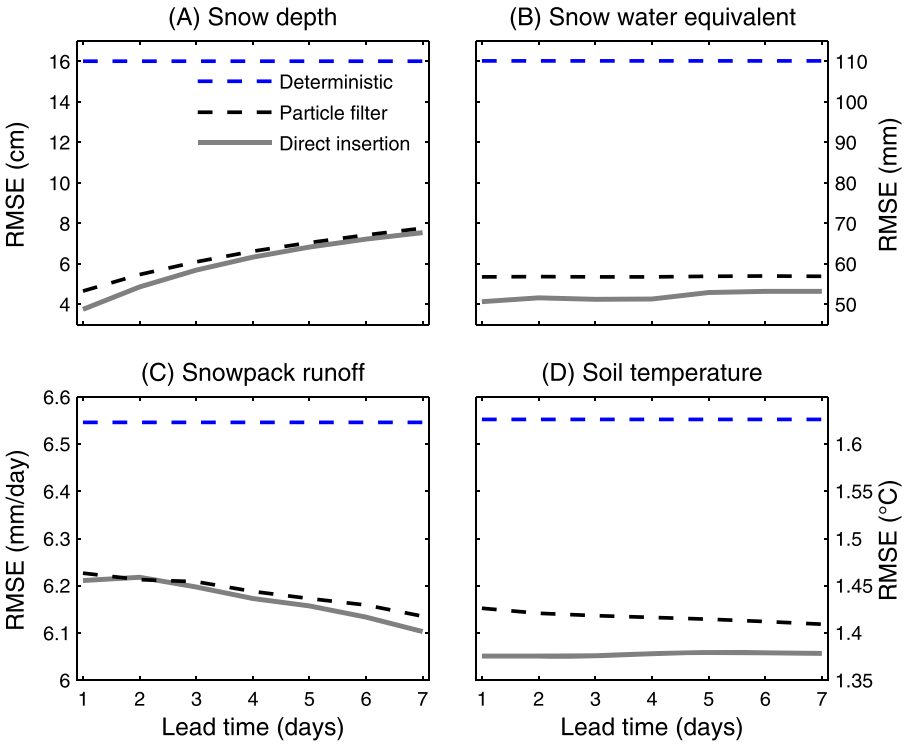

Figure 9. Performance of the snow model for varying lead-times for the deterministic, particle filter and direct insertion simulations at Col de Porte using SAFRAN input data. The particle filter uses 2000 ensemble members, and also perturbations on the snowfall correction factor (see section 4.2). The evaluation covers the complete study period from 1 October 1994 to 30 June 2010.

errors, or (3) the true parameter value is outside of the prior range of the parameter. Nevertheless, the particle filter seems able to estimate the snowfall correction factor robustly, and may after further testing, be a useful approach for reducing the uncertainty in one of the most sensitive forcing variables for snowpack simulations.

5.7. Predictive Performance at Col de Porte

We assessed the predictive performance of the snow model in combination with the particle filter and direct insertion for varying lead-times using the SAFRAN data set as forcing. To reduce the influence of any biases in the precipitation input data during the forecasting period, we use the same setup for the particle filter as in the combined state and parameter estimation experiment (see section 4.2). For snow depth and SWE (Figures 9a and 9b), we find that the RMSE is substantially lower for both the particle filter and direct insertion than the error of the deterministic run for the complete forecasting period. For snow depth (Figure 9a), the RMSE increases slightly with lead-time for both data assimilation methods. For SWE and soil temperature (Figures 9b and 9d), the RMSE does not change much over the forecasting period for direct insertion as well as the particle filter. Finally, for snowpack runoff, the RMSE even improves slightly with lead-time for both data assimilation methods. Overall, the difference between the both data assimilation methods is small. For all variables, the small changes in RMSE with lead-time depends on the strong temporal autocorrelation of snowpack variables. Thus, improvements of the snow simulations by the particle filter or direct insertion remain visible for the complete forecasting period.

5.8. Results From Field Sites in Switzerland

As an independent test of the achievements reported from Col de Port, we ran the same simulations for 40 field sites in Switzerland. The particle filter was run using 2000 ensemble members to avoid eventual filter degeneracy and with perturbations representing random errors in the input data (see section 4.1). The particle filter algorithm efficiently reduces the errors in simulated snow depth (compare Figure 10a and 10b) and SWE (see Figures 10d and 10e). The deterministic simulations typically overestimate both variables, indicating biased input data. Note that for the purpose of this test on the capabilities of data assimilation

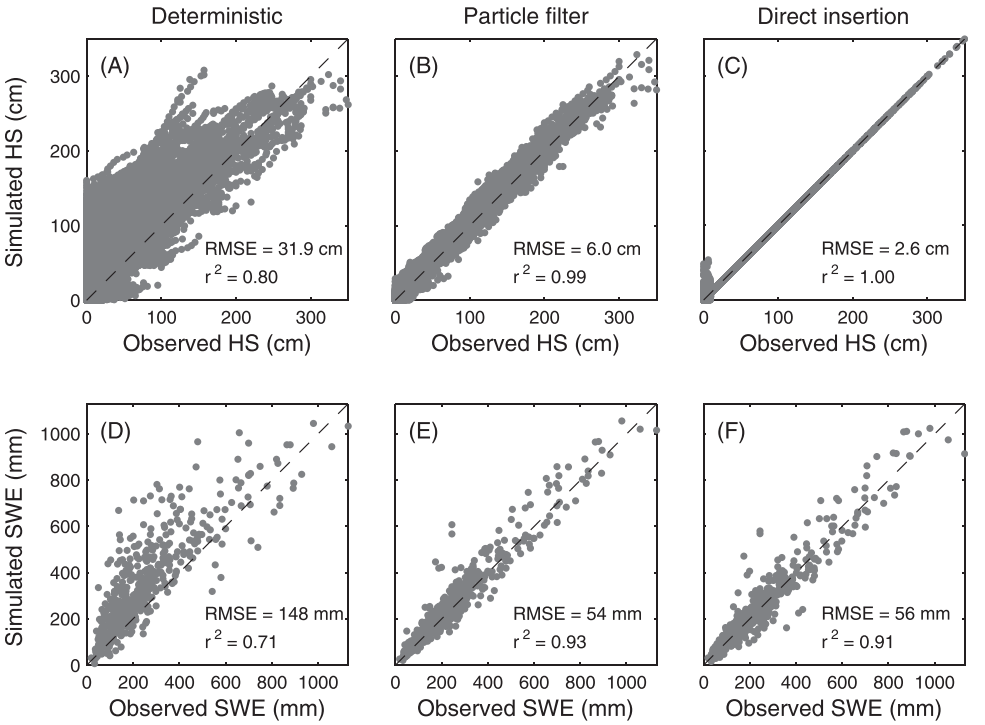

Figure 10. Scatterplots showing the performance of (a and d) the deterministic simulations, (b and e) the particle filter algorithm, and (c and f) direct insertion for snow depth (HS) and snow water equivalent (SWE). The evaluation period spans from 10 September 2013 to 31 July 2015 and comprises 24,320 snow depth and 503 SWE observations. For the particle filter results, the figure shows the mean value of the ensemble.

methods, a known systematic cold bias in the temperature input data has not been corrected in preprocessing. The cold bias limits melt and increases the proportion of snowfall relative to rain. At many of the stations, this bias likely causes the large overestimation of snow depth and SWE (Figures 10a and 10d). Although the presence of systematic errors in several of the forcing data variables were not accounted for in the perturbations, the particle filter still markedly improved model performance.

For snow depth, the particle filter reduces the RMSE by approximately 81%, whereas the corresponding reduction in error for SWE equals 64%. The slightly lower reduction of the latter variable can be due to the larger errors associated with measured SWE than snow depth, errors in simulated snow density and the difference in location for the SWE and snow depth observations. For SWE, direct insertion shows slightly higher RMSE (∼4%) and lower squared correlation coefficient (∼2%) than the particle filter results (compare Figures 10e and 10f). Obviously, for the assimilation of snow depths, whether using the particle filter or direct insertion, it is critically important that the model simulate snow density correctly. Thus, if using the data assimilation strategy proposed in this study with another snow model, it is necessary to evaluate the ability of the model to predict snow density reliably. We suggest to identify an appropriate snow model for a specific region by using a multimodel framework such as that proposed by *Essery et al.* [2013] and tested in *Magnusson et al.* [2015].

5.9. Difference in Behavior Between Direct Insertion and Particle Filter

With the following simple example, we demonstrate contrasts in how the direct insertion and particle filter methods handle different situations. These differences are related to choices that must be made when using direct insertion that predetermine how model dynamics are affected when assimilating data. In our application, decreases of SWE introduced by direct insertion were removed from the snowpack without contributing to runoff while increases of SWE took on the current modeled density and temperature of the surface layer. The particle filter method, on the other hand, inherently produces model states consistent with the introduced changes and tends toward more conservative corrections since observations are not considered error free.

Figure 11 shows the simulation results from Col de Porte for the 2003/2004 winter in which the deterministic model underpredicts observed snow depths. For two periods, one in early winter (period 1) and one in mid-winter (first half of period 2), snowfall events are followed by a period of settling without pronounced melt. For both periods, the deterministic simulations predict too slow settling, and direct insertion compensates for this mismatch by removing snow (SWE decreases), whereas the particle filter favors particles with faster settling rates (SWE remains constant). Both data assimilation methods produced similar runoff estimates matching the runoff observations well for both periods. For direct insertion, the predetermined decision to not have removed snow contribute to runoff was correct in this case, but may lead to erroneous results in other instances. Predefined rules such as this can introduce physically incorrect changes in the model states (e.g., the violation of mass conservation or a static density while the snow settles). The particle filter, on the other hand, inherently keeps the physical relationship realistic between model states and the assimilated data. Furthermore, for the second half of period 2, the deterministic run seems to overestimate settling rates. In this case, direct insertion adds snow even though observed depth decreases and SWE remains rather stable. Though the particle filter simulated snow depths deviate slightly from the observations, the simulated SWE is in greater accordance with the theoretical expectations of a settling snowpack.

To summarize the discussion, direct insertion (a) often requires model-specific considerations, (b) assumes observations free from errors, (c) may violate physical principles when a priori modeling choices do not match encountered conditions, and (d) is therefore challenging to implement. The more generic particle filter, on the other hand, avoids many of those limitations without negative impacts on the performance (see results in sections 5.2–5.8).

6. Summary and Conclusions

In this study, we show that the assimilation of daily snow depths using the particle filter improves the results of a multilayer energy-balance snow model. After assimilation, the snow model closely tracks the observed snow depths, and greatly improves estimates of SWE. While the assimilation algorithm improves simulated total runoff over the entire study period, model-predicted daily runoff dynamics did not substantially benefit from the snow depth assimilation. Limitations of the snow model to accurately compute daily snowmelt, rainfall-driven runoff events or uncertainties in the lysimeter observations may mask any improvements obtained by the assimilation of snow depth for daily runoff. However, during some important periods, for example during the melt-out period, the assimilation of snow depths seems to improve the runoff predictions (see Figure 3). For soil temperature, the particle filter improves the results mainly during spring due to a more accurate simulation of the snow disappearance date. Thus, as observed for several variables, the assimilation of snow depths improved simulated states and fluxes, including simulated properties of the underlying soil column.

In many situations, snow models give poor results due to biases in the forcing data [*Raleigh et al*., 2015]. With the particle filter algorithm, we could estimate biases in snowfall rates using the snow depth data in a combined state and parameter estimation experiment. In future studies, it might be worthwhile to test whether we can estimate the bias in several forcing variables simultaneously using more sophisticated data assimilation algorithms, or by assimilating several observed properties simultaneously.

The assimilation of snow depths improved the model results, in particular when the model was forced with coarse resolution data, from either the SAFRAN system or a weather forecasting model. For some variables, such as SWE, particle filter simulations forced with poor-quality data were at times comparable to nonassimilated model outcomes driven by high-quality data. Thus, inexpensive measurements of snow depth along with the particle filter algorithm forced with coarse meteorological model data (e.g., from a weather forecast) were able to produce results of comparable quality to simulations driven with data from very expensive field installations.

For purposes of operational modeling and snowmelt forecasting, it is particularly noteworthy that the improvements in model performance achieved with data assimilation when input data exhibited biases were not achieved at the expense of model performance when input data were excellent. The quality of input data from a weather forecast is highly variable and might be severely biased one day, but perfect for another day. In such a setting, it is mandatory that data assimilation procedures are robust and can deal

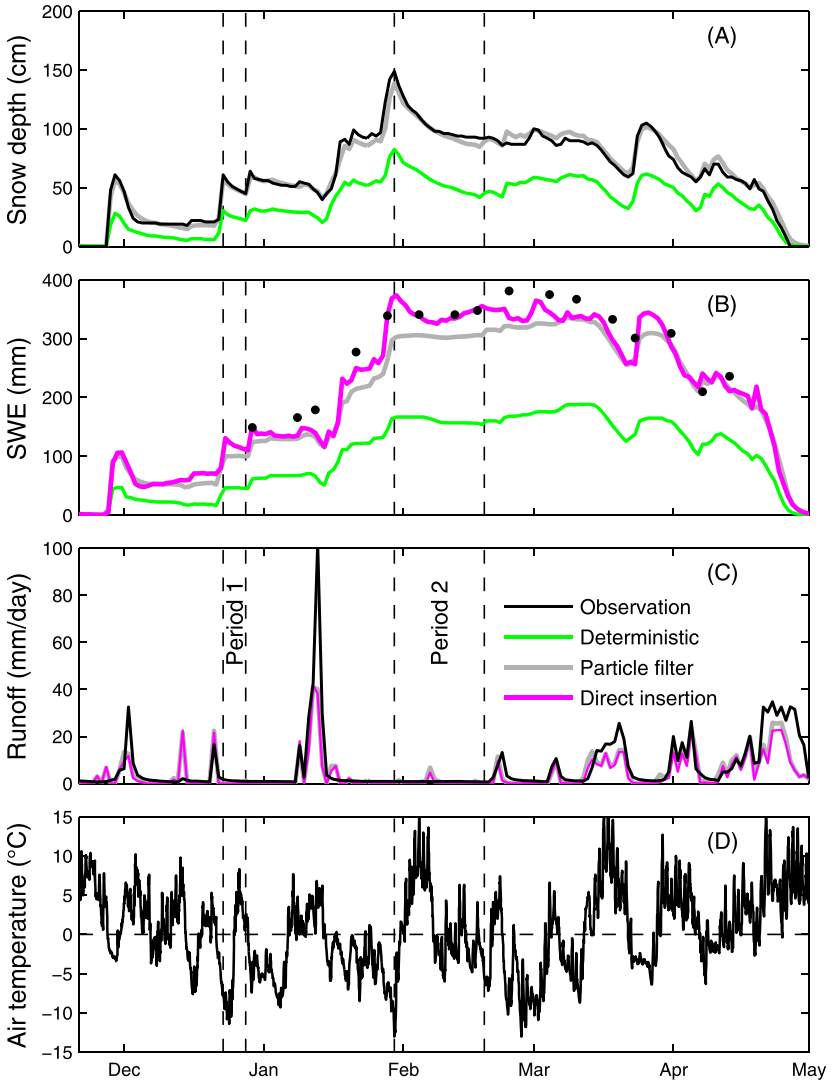

Figure 11. Simulated and observed snow depth, SWE, and snowpack runoff at Col de Porte for the 2003/2004 winter. The snow depths obtained for the direct insertion method are not shown since they match the observations as long as depths exceed 10 cm. The particle filter results (2000 ensemble members) were produced using the combined state and parameter estimation strategy (see section 4.2), and are presented using the expected value of the ensemble. The air temperatures were taken from the in situ measurements.

with both scenarios. Also, dynamic data assimilation, accounting for observed data in hindcast and running on weather model data in forecast, would likely benefit from the particle filter approach.

The particle filter and direct insertion method showed similar performance for reproducing SWE, snowpack runoff and soil temperature. At first, the conceptually simple direct insertion scheme seems like a more attractive method than the particle filter. However, implementing direct insertion in a complex, multilayered snow model is not straightforward. Deciding how state variables should be modified based on differences in a bulk snowpack observation such as snow depth and implementing these changes within the model requires a complete understanding of model functionality and coding. The particle filter is typically very easy to implement with most snow models. More important, even though the direct insertion method performed well, the scheme produced inconsistencies between modeled states. The particle filter, on the other hand, avoids those limitations without significant loss in performance. Finally, the ensemble-based method will be more robust to errors in the model and observational data since it accounts for these errors whereas the direct insertion method does not.

In a recent study, *Margulis et al.* [2015] assimilated satellite observations of snow cover fraction into a land surface model using a particle smoother. Their data assimilation method reduced RMSE for

reconstructed SWE simulations by 60–82% compared to the unassimilated simulations. In this study, we find corresponding reductions in RMSE of 47–64% dependent on location and forcing data set. The design of the Margulis et al. application, however, is not suitable for flood forecasting applications such as the one presented in this research. Other studies of snow data assimilation typically show similar or lower improvements than those reported above [e.g., *De Lannoy et al*., 2012; *Liu et al*., 2013]. However, the value of these types of intercomparisons are questionable due to potentially large differences in the performance of the unassimilated simulations, assimilation methods, quality of forcing and evaluation data, type of application, and local conditions. A comprehensive evaluation of different data assimilation methods using common snow models and data sets would be valuable, similarly as has been done for snow models [e.g., *Slater et al*., 2001].

Many recent studies focus on developing methods for assimilating remotely sensed snowpack properties, such as SWE, reflectance data, and snow cover fraction [e.g., *Andreadis and Lettenmaier*, 2006; *Charrois et al*., 2015; *De Lannoy et al*., 2012; *Liu et al*., 2013]. While SWE is of particular importance to hydrologists, automated SWE measurements are costly and except for the western United States rarely available. Manual observations are also costly and by their nature, spatially and temporally limited. Furthermore, though substantial effort has been invested in remote sensing SWE retrievals, these data remain low-resolution products with high uncertainties [*Tong and Velicogna*, 2010; *Byun and Choi*, 2014]. By contrast, instruments for regularly measuring snow depth are inexpensive and readily available, while modern LiDAR technologies, though costly, are currently providing high-resolution, accurate snow depth data over ever-increasing areas [*Egli et al*., 2012; *Painter et al*., 2016]. In this study, we have shown the benefit of utilizing daily snow depth data in physically based point snow models. Future work will test these methods in spatially distributed models in regions with regularly scheduled LiDAR snow retrievals. This will allow us to upscale the findings from this study to larger basins, and test whether the method improves variables such as streamflow predictions. Finally, data assimilation methods based on Bayesian statistics such as the particle filter require reliable estimates of the uncertainties in the forcing and assimilation data, as well as in the model itself. For snow models, more work is still needed for better representing those uncertainties in the data assimilation setup.

Acknowledgment
We thank Marie Dumont and two additional reviewers for their helpful comments on this paper and Samuel Morin for providing information about the Col de Porte data set. This study was partly funded by the Federal Office of the Environment FOEN. Data used in this study can be made available upon request from the main author (e-mail address: jmg@nve.no).

References

Anderson, E. A. (1976), A point energy and mass balance model of a snow cover, *NOAA Tech. Rep. NWS* 19, 150 pp., Off. of Hydrol., Natl. Weather Serv., Silver Spring, Md.

Andreadis, K. M., and D. P. Lettenmaier (2006), Assimilating remotely sensed snow observations into a macroscale hydrology model, *Adv. Water Resour.*, *29*(6), 872–886, doi:10.1016/j.advwatres.2005.08.004.

Arulampalam, M. S., S. Maskell, N. Gordon, and T. Clapp (2002), A tutorial on particle filters for on-line non- linear/non-Gausssian Bayesin tracking, *IEEE Trans. Signal Process*, *50*(2), 174–188.

Bartelt, P., and M. Lehning (2002), A physical SNOWPACK model for the Swiss avalanche warning Part I: Numerical model, *Cold Reg. Sci. Technol.*, *35*(3), 123–145, doi:10.1016/S0165-232X(02)00074-5.

Boone, A., and P. Etchevers (2001), An intercomparison of three snow schemes of varying complexity coupled to the same land surface model: Local-scale evaluation at an Alpine Site, *J. Hydrometeorol.*, *2*(4), 374–394, doi:10.1175/1525-7541(2001)002 < 0374:AIOTSS>2.0.CO;2.

Brown, R. D., B. Brasnett, and D. Robinson (2003), Gridded North American monthly snow depth and snow water equivalent for GCM evaluation, *Atmos. Ocean*, *41*(1), 1–14, doi:10.3137/ao.410101.

Brun, E., P. David, M. Sudul, and G. Brunot (1992), A numerical model to simulate snow-cover stratigraphy for operational avalanche forecasting, *J. Glaciol.*, *38*(128), 13–22.

Byun, K., and M. Choi (2014), Uncertainty of snow water equivalent retrieved from AMSR-E brightness temperature in northeast Asia, *Hydrol. Processes*, *28*(7), 3173–3184, doi:10.1002/hyp.9846.

Charrois, L., E. Cosme, M. Dumont, M. Lafaysse, S. Morin, Q. Libois, and G. Picard (2015), On the assimilation of optical reflectances and snow depth observations into a detailed snowpack model, *Cryosph. Discuss.*, *9*(6), 6829–6870, doi:10.5194/tcd-9-6829-2015.

Dechant, C., and H. Moradkhani (2011), Radiance data assimilation for operational snow and streamflow forecasting, *Adv. Water Resour.*, *34*(3), 351–364, doi:10.1016/j.advwatres.2010.12.009.

De Lannoy, G. J. M., R. H. Reichle, K. R. Arsenault, P. R. Houser, S. Kumar, N. E. C. Verhoest, and V. R. N. Pauwels (2012), Multiscale assimilation of Advanced Microwave Scanning Radiometer–EOS snow water equivalent and Moderate Resolution Imaging Spectroradiometer snow cover fraction observations in northern Colorado, *Water Resour. Res.*, *48*, W01522, doi:10.1029/2011WR010588.

Douc, R., O. Cappe, and E. Moulines (2005), Comparison of resampling schemes for particle filtering, in *ISPA 2005: Proceedings of 4th International Symposium on Image Signal Processing and Analysis 2005*, pp. 64–69, IEEE, New York, doi:10.1109/ISPA.2005.195385.

Doucet, A., N. De Freitas, and N. Gordon (2001), *Sequential Monte Carlo Methods in Practice*, Springer, New York.

Douville, H., J. Royer, and J. Mahfouf (1995), A new snow parameterization for the Meteo-France climate model. Part I: Validation in stand-alone experiments, *Clim. Dyn.*, *12*, 21–35, doi:10.1007/BF00208760.

Durand, M., E. J. Kim, and S. A. Margulis (2009), Radiance assimilation shows promise for snowpack characterization, *Geophys. Res. Lett.*, *36*, L02503, doi:10.1029/2008GL035214.

Durand, Y., E. Brun, L. Merindol, G. Guyomarc'h, B. Lesaffre, and E. Martin (1993), A meteorological estimation of relevant parameters for snow models, *Ann. Glaciol.*, *18*, 65–71.

Egli, L., T. Jonas, T. Grünewald, M. Schirmer, and P. Burlando (2012), Dynamics of snow ablation in a small Alpine catchment observed by repeated terrestrial laser scans, *Hydrol. Processes*, *26*(10), 1574–1585, doi:10.1002/hyp.8244.

Essery, R., S. Morin, Y. Lejeune, and C. B. Menard (2013), A comparison of 1701 snow models using observations from an alpine site, *Adv. Water Resour.*, *55*, 131–148, doi:10.1016/j.advwatres.2012.07.013.

Evensen, G. (2003), The Ensemble Kalman Filter: Theoretical formulation and practical implementation, *Ocean Dyn.*, *53*(4), 343–367, doi:10.1007/s10236-003-0036-9.

Fletcher, S. J., G. E. Liston, C. A. Hiemstra, and S. D. Miller (2012), Assimilating MODIS and AMSR-E snow observations in a snow evolution model, *J. Hydrometeorol.*, *13*(5), 1475–1492, doi:10.1175/jhm-d-11-082.1.

Kantas, N., A. Doucet, S. S. Singh, J. Maciejowski, and N. Chopin (2015), On particle methods for parameter estimation in general state-space models, *Stat. Sci.*, *30*(3), 328–351, doi:10.1214/14-STS511.

Kavetski, D., and G. Kuczera (2007), Model smoothing strategies to remove microscale discontinuities and spurious secondary optima in objective functions in hydrological calibration, *Water Resour. Res.*, *43*, W03411, doi:10.1029/2006WR005195.

Kumar, M., D. Marks, J. Dozier, M. Reba, and A. Winstral (2013), Evaluation of distributed hydrologic impacts of temperature-index and energy-based snow models, *Adv. Water Resour.*, *56*, 77–89, doi:10.1016/j.advwatres.2013.03.006.

Leisenring, M., and H. Moradkhani (2011), Snow water equivalent prediction using Bayesian data assimilation methods, *Stochastic Environ. Res. Risk Assess.*, *25*(2), 253–270, doi:10.1007/s00477-010-0445-5.

Leys, C., C. Ley, O. Klein, P. Bernard, and L. Licata (2013), Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median, *J. Exp. Soc. Psychol.*, *49*(4), 764–766, doi:10.1016/j.jesp.2013.03.013.

Liu, Y., C. D. Peters-Lidard, S. Kumar, J. L. Foster, M. Shaw, Y. Tian, and G. M. Fall (2013), Assimilating satellite-based snow depth and snow cover products for improving snow predictions in Alaska, *Adv. Water Resour.*, *54*, 208–227, doi:10.1016/j.advwatres.2013.02.005.

Lopez-Moreno, J. I., S. R. Fassnacht, S. Begueria, and J. B. P. Latron (2011), Variability of snow depth at the plot scale: Implications for mean depth estimation and sampling strategies, *Cryosphere*, *5*(3), 617–629, doi:10.5194/tc-5-617-2011.

Louis, J. F. (1979), A parametric model of vertical eddy fluxes in the atmosphere, *Boundary Layer Meteorol.*, *17*(2), 187–202, doi:10.1007/BF00117978.

Luetschg, M., M. Lehning, and W. Haeberli (2008), A sensitivity study of factors influencing warm/thin permafrost in the Swiss Alps, *J. Glaciol.*, *54*(187), 696–704.

Magnusson, J., D. Gustafsson, F. Hüsler, and T. Jonas (2014), Assimilation of point SWE data into a distributed snow cover model comparing two contrasting methods, *Water Resour. Res.*, *50*, 7816–7835, doi:10.1002/2014WR015302.

Magnusson, J., N. Wever, R. Essery, N. Helbig, A. Winstral, and T. Jonas (2015), Evaluating snow models with varying process representations for hydrological applications, *Water Resour. Res.*, *51*, 2707–2723, doi:10.1002/2014WR016498.

Margulis, S. A., M. Girotto, G. Cortés, and M. Durand (2015), A particle batch smoother approach to snow water equivalent estimation, *J. Hydrometeorol.*, *16*, 1752–1772, doi:10.1175/JHM-D-14-0177.1.

Margulis, S. A., G. Cortés, M. Girotto, and M. Durand (2016), A Landsat-era Sierra Nevada (USA) snow reanalysis (1985-2015), *J. Hydrometeorol.*, *17*, 1203–1221, doi:10.1175/JHM-D-15-0177.1.

McGuire, M., A. Wood, A. Hamlet, and D. Lettenmaier (2006), Use of satellite data for streamflow and reservoir storage forecasts in the Snake River Basin, *J. Water Resour. Plann. Manage.*, *132*(2), 97–110, doi:10.1061/(ASCE)0733-9496(2006)132:2(97).

Morin, S., Y. Lejeune, B. Lesaffre, J.-M. Panel, D. Poncet, P. David, and M. Sudul (2012), A 18-yr long (1993–2011) snow and meteorological dataset from a mid-altitude mountain site (Col de Porte, France, 1325 m alt.) for driving and evaluating snowpack models, *Earth Syst. Sci. Data*, *4*, 13–21, doi:10.5194/essdd-5-29-2012.

Oleson, K. W., et al. (2004), Technical description of the Community Land Model (CLM), *NCAR Tech. Note NCAR/TN-461+STR*, Natl. Cent. for Atmos. Res., Boulder, Colo.

Painter, T. H., et al. (2016), The airborne snow observatory: Fusion of scanning Lidar, imaging spectrometer, and physically-based modeling for mapping snow water equivalent and snow albedo, *Remote Sens. Environ.*, *184*, 139–152, doi:10.1016/j.rse.2016.06.018.

Raleigh, M. S., J. D. Lundquist, and M. P. Clark (2015), Exploring the impact of forcing error characteristics on physically based snow simulations within a global sensitivity analysis framework, *Hydrol. Earth Syst. Sci.*, *19*(7), 3153–3179, doi:10.5194/hess-19-3153-2015.

Rössler, O., P. Froidevaux, U. Börst, R. Rickli, O. Martius, and R. Weingartner (2014), Retrospective analysis of a nonforecasted rain-on-snow flood in the Alps—A matter of model limitations or unpredictable nature?, *Hydrol. Earth Syst. Sci.*, *18*(6), 2265–2285, doi:10.5194/hess-18-2265-2014.

Shrestha, M., L. Wang, T. Koike, Y. Xue, and Y. Hirabayashi (2010), Improving the snow physics of WEB-DHM and its point evaluation at the SnowMIP sites, *Hydrol. Earth Syst. Sci.*, *14*(12), 2577–2594, doi:10.5194/hess-14-2577-2010.

Slater, A. G., and M. P. Clark (2006), Snow data assimilation via an ensemble Kalman filter, *J. Hydrometeorol.*, *7*(3), 478–493, doi:10.1175/JHM505.1.

Slater, A. G., et al. (2001), The representation of snow in land surface schemes: Results from PILPS 2(d), *J. Hydrometeorol.*, *2*(1), 7–25, doi:10.1175/1525-7541(2001)002 < 0007:TROSIL>2.0.CO;2.

Thirel, G., P. Salamon, P. Burek, and M. Kalas (2013), Assimilation of MODIS snow cover area data in a distributed hydrological model using the particle filter, *Remote Sens.*, *5*(11), 5825–5850, doi:10.3390/rs5115825.

Tong, J., and I. Velicogna (2010), A comparison of AMSR-E/Aqua snow products with in situ observations and MODIS snow cover products in the Mackenzie River Basin, Canada, *Remote Sens.*, *2*(10), 2313–2322, doi:10.3390/rs2102313.

Vionnet, V., E. Brun, S. Morin, A. Boone, S. Faroux, P. Le Moigne, E. Martin, and J. M. Willemet (2012), The detailed snowpack scheme Crocus and its implementation in SURFEX v7.2, *Geosci. Model Dev.*, *5*(3), 773–791, doi:10.5194/gmd-5-773-2012.

Wever, N., C. Fierz, C. Mitterer, H. Hirashima, and M. Lehning (2014), Solving Richards Equation for snow improves snowpack meltwater runoff estimations in detailed multi-layer snowpack model, *Cryosphere*, *8*(1), 257–274, doi:10.5194/tc-8-257-2014.

Zanotti, F., S. Endrizzi, G. Bertoldi, and R. Rigon (2004), The GEOTOP snow module, *Hydrol. Processes*, *18*(18), 3667–3679, doi:10.1002/hyp.5794.

Zhao, Q., Z. Liu, B. Ye, Y. Qin, Z. Wei, and S. Fang (2009), A snowmelt runoff forecasting model coupling WRF and DHSVM, *Hydrol. Earth Syst. Sci. Discuss.*, *13*, 1897–1906, doi:10.5194/hessd-6-3335-2009.