Version 3.0.1 of the Crocus snowpack model

Matthieu Lafaysse[1], Marie Dumont[1], Basile De Fleurian[1], Mathieu Fructus[1], Rafife Nheili[1], Léo Viallon-Galinier[1], Matthieu Baron[1], Aaron Boone[2], Axel Bouchet[1], Julien Brondex[1,4], Carlo Carmagnola[1], Bertrand Cluzet[1], Kévin Fourteau[1], Ange Haddjeri[1,4], Pascal Hagenmuller[1], Giulia Mazzotti[1,4], Marie Minvielle[2], Samuel Morin[1], Louis Quéno[1,5], Léon Roussel[1], Pierre Spandre[1], François Tuzet[1], and Vincent Vionnet[1,3]

[1]Météo-France, CNRS, Univ. Grenoble Alpes, Univ. Toulouse, CNRM, Centre d'Études de la Neige, Grenoble, France
[2]Météo-France, CNRS, Univ. Toulouse, CNRM, Toulouse, France
[3]Meteorological Research Division, Environment and Climate Change Canada, Dorval, QC, Canada
[4]now at Institut des Géosciences de l'Environnement (IGE), CNRS, INRAE, IRD, Grenoble INP, Univ. Grenoble Alpes, Grenoble, France
[5]now at WSL Institute for Snow and Avalanche Research (SLF), Davos, Switzerland

Correspondence: Matthieu Lafaysse (matthieu.lafaysse@meteo.fr)

Abstract. This article presents a comprehensive description of the 3.0.1 stable release of the Crocus snowpack model in the SURFEX modelling platform. It synthesizes and harmonizes a number of equations disseminated in various previous publications, introduces a number of unpublished parameterizations and includes new developments implemented since 2012. Among the novelties, an explicit representation of the evolution of impurity mass in snow (e.g. black carbon, mineral dust) allows representing their impact on solar radiation absorption in the snowpack at different wavelengths and their feedback on all snowpack properties. The model also allows the formation of surface ice layers due to freezing rain. In addition, Crocus is coupled to the MEB "big-leaf" vegetation scheme and can therefore be applied in forested areas. A module for snow management can also be optionally activated to simulate the snowpack on ski slopes in ski resorts. The model can be coupled with various blowing snow schemes. The MEPRA expert system which analyses the mechanical stability of the simulated snowpack has been implemented directly within SURFEX. ~~A multiphysics version of the model (ESCROC) was also developed by implementing from 2 to 4 parameterizations from the literature for~~ For each physical process represented by empirical parameterizations, several new parameterizations from the literature were implemented. The different combinations ~~allow~~ of these parameterizations constitute the ESCROC multiphysics ensemble model. It allows the quantification of simulations uncertainty for various applications. Finally, a technical solution was proposed for externalized applications allowing the use of the scheme in other Land Surface Models. The paper also reviews the available scientific evaluations and applications of the model. It describes its numerical efficiency and the main scientific and technical challenges providing guidance for the future of snow modelling.

1 Introduction

A large variety of snowpack modelling systems (e.g. **?**) have been developed for several decades for various applications (**?**): the computation of energy fluxes at the atmosphere-cryosphere interface in climate modelling and numerical weather prediction; hydrological simulations for discharge forecasting, water resources and hydropower management; physical process studies in the snowpack; avalanche hazard forecasting; glacier mass balance assessment; sea ice modelling; etc. The most detailed snowpack models include a detailed representation of the snowpack stratigraphy as well as an explicit representation of some microstructural properties of the snow layers through the implementation of empirical parameterization of snow metamorphism. There is a relatively limited number of snowpack models with such a level of detail: mainly the "Swiss" SNOWPACK model (**?**), the "American" SNTHERM model (**?**), the "Japanese" SMAP model (**?**) and the "French" Crocus model. The latter has been initially developed by **??** and was implemented in the early 2010s in the SURFEX platform of surface modelling (**?**) to facilitate coupling with atmospheric models and the other components of surface modelling, especially soil and vegetation schemes of the ISBA Land Surface Model. The last description of the model was published by **?** after this major evolution.

However, numerous evolutions of the model have been implemented after **?**. Some of them are only partly described in dedicated scientific publications (**??????**). These articles were generally published before the merging of all new developments in a unique and stable code version. A comprehensive and accurate description of the state of the last official model release is therefore missing. This lack of documentation has been identified by **?** as one of the main factors of human errors in numerical simulations. The purpose of this paper is to provide an updated reference with all the functionalities available in the latest stable release of the model. To help the users to find any information they may need to understand the model implementation, all the equations disseminated in the various papers are reported here either in the main text or in Appendices, including the equations already published in previous references of **?** and **?** in order to provide a self-sufficient reference describing the whole model. Following the recommendations of **?**, a major effort has been dedicated to check the consistency and comprehensiveness between the code and the equations. The scope of this paper is limited to the snowpack on the ground. The coupled modules representing soil, vegetation and blowing snow remain beyond the scope of this paper and only the terms involved in the coupling are mentioned. More details can be found in the reference publications of the coupled models: **??** for soil and vegetation; **??** and **?** for blowing snow, cf. Section 2.4.2. Note that the standalone Crocus model was last designated as version 2.4. This versioning was discontinued by **?** in favor of the SURFEX versioning system. However, due to the impossibility to synchronize Crocus and SURFEX main stable releases and the integration of Crocus in other Land Surface Models, a dedicated versioning is necessary for Crocus. This paper thus describes version 3.0.1 of Crocus. Section 2 presents all scientific equations necessary to ~~make~~ evolve the state variables of the model. Section 3 presents complementary diagnoses computed as output of the model. Section 4 presents technical features associated with running the model (simulation geometries, numerical efficiency, and associated tools to facilitate running and visualization). Finally, Section 5 reviews the available scientific evaluations of the model and provides an overview of its applications~~, to discuss the associated confidence~~

Table 1. Available physical options in Crocus. Default options appear in bold. The last column shows options which are not available in the externalized release but only within the SURFEX implementation of Crocus. Although the combination of all options is possible in theory, the options for snow management in ski resorts were only tested with the default options of the other processes and the blowing snow schemes were never tested together with the MEB vegetation scheme.

Process	Namelist key	Options	Requirements	Only in SURFEX
Snowfall	SNOWFALL	**V12**, S14, A76		
Metamorphism	SNOWMETAMO	**B21**, F06, S-B, S-F		
Compaction	SNOWCOMP	**B92**, T11, S14		
Drifting snow (subgrid effect)	SNOWDRIFT	NONE, DFLT, **VI13**, GA01		
Mobility index	SNOWMOB	**GM98**, VI12		
Absorption of solar radiation	SNOWRAD	**B92**, T17	T17 requires LAP forcing	
Thermal conductivity	SNOWCOND	**Y81**, C11, I02		
Percolation	SNOWLIQ	**B92**, SPK, B02		
Snow-vegetation interactions	MEB	True, **False**		X
Blowing snow (transport)	SNOWPAPPUS	True, **False**	2-D grids	X
Blowing snow (transport)	SNOWSYTRON	True, **False**	4 or 8 slope aspects	X
Machine-made snow	SNOWMAK_BOOL	True, **False**		X
Production strategy	SELF_PROD	True, **False**	Water use target	X
Grooming	SNOWCOMPACT	True, **False**		X
Tilling	SNOWTILLER	True, **False**		X

~~and~~. In light of these elements, we discuss the confidence that can be placed in the simulation results and the main remaining challenges for the future, including perspectives in terms of data assimilation.

2 Physical model

Compared to **?**, the model has become highly modular with the extension of applications and the development of ~~a~~ the ESCROC multiphysics framework by **?**. The common modelling structure includes the discretization procedure and the solving of the heat diffusion equation which is the core of the model. Figure 1 presents an overview of the sequence of routines and summarizes the main changes (new routines and updated routines) which are detailed in this Section. Table 1 summarizes the different available options for each process and associated requirements. The details of each option are described in the subsection dedicated to the corresponding process. The ESCROC multiphysics ensemble system consists in the combination of these differents physical options and different values of some key parameters such as τ_a (Section 2.4.9), Ri_l (Section 2.4.11).

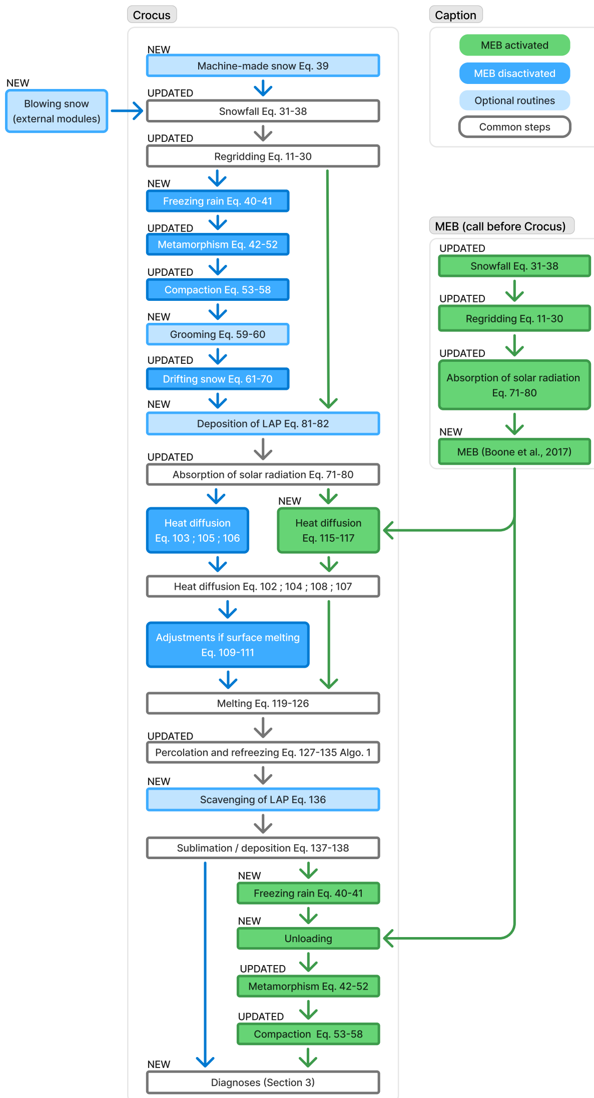

Figure 1. Sequence of Crocus subroutines for each time step without (blue) or with (green) coupling with MEB. Optional subroutines in light blue. Compared to the version of **?**, new subroutines are labelled "NEW". The label "UPDATED" refers to subroutines in which changes were applied due to the new state variables for microstructure and where new physical options were implemented. The heat diffusion, melting and sublimation routines were only concerned by minor changes or technical optimizations.

2.1 Forcing variables

The model has to be forced with subdaily time series of air temperature T_a (K) and specific humidity q_a ($\mathrm{kg\,kg^{-1}}$) at a known level z_a (m), wind speed U ($\mathrm{m\,s^{-1}}$) at a known level z_u (m), incoming longwave radiation LW↓ ($\mathrm{W\,m^{-2}}$), incoming shortwave direct and diffuse radiation $\mathrm{SW_{DIR}}$↓ and $\mathrm{SW_{DIF}}$↓ ($\mathrm{W\,m^{-2}}$), rainfall $\mathcal{P}_r$ and snowfall $\mathcal{P}_s$ ($\mathrm{kg\,m^{-2}\,s^{-1}}$) and surface pressure P_s (Pa). The very low sensitivity of model results to P_s allows the user to provide constant P_s values when a time series is not available. The split of global shortwave radiation SW↓ between direct and diffuse components is only used with specific model options (coupling with TARTES optical scheme, Section 2.4.9, and/or coupling with MEB big-leaf vegetation scheme, Section 2.4.14). When none of these options are ~~not activated, a random~~ activated, the split between both components is ~~sufficient~~ not necessary as only their sum is considered. ~~For~~ When TARTES or MEB are activated, an accurate forcing of $\mathrm{SW_{DIR}}$↓ and $\mathrm{SW_{DIF}}$↓ should always be preferred. Nevertheless, for users interested in these options without available data to separate both components, the following parameterization as a function of the cosine of the solar zenith angle μ and derived from SBDART clear-sky modelling (**?**) at Col de Porte is ~~reasonable in~~ available for mid-latitude areas~~and~~. Following **?**, the estimated error in cloudy conditions for the simulated broadband albedo is lower than 0.1. The parameterization is applied by default in the code if the diffuse ~~components~~ component is set to zero:

$$\frac{\mathrm{SW_{DIF}}}{\mathrm{SW_{DIR}}+\mathrm{SW_{DIF}}} = \min\left(\exp\left(-1.549919\mu^3 + 3.735357\mu^2 - 3.524211\mu + 0.029911\right), 1.\right) \tag{1}$$

Optionally, dry and wet deposition fluxes of Light-Absorbing Particles (LAP) $\mathcal{W}_{d,j}$ and $\mathcal{W}_{w,j}$ ($\mathrm{kg\,m^{-2}\,s^{-1}}$) can be included for each type j depending on the physical option selected for radiative transfer (Section 2.4.9).

2.2 State variables

The prognostic variables (i.e. transmitted from a given time step to the next one) describing each snow layer i are its mass m_i ($\mathrm{kg\,m^{-2}}$), density ρ_i ($\mathrm{kg\,m^{-3}}$), enthalpy H_i ($\mathrm{J\,m^{-2}}$) defined as the energy required to melt the snow layer i (**?**), age A_i (days since snowfall), and complementary variables for snow microstructure. The state variables initially used for snow microstructure as described in **?** (dendricity, sphericity, grain size) were replaced by optical diameter d_i (m) and sphericity $S_i \in [0,1]$ (**?**). After several years of coexistence, the formulation from **?** was removed from the code in order to improve its readability but also its efficiency as numerous expensive conditional statements could be removed. This change of state variables does not only affect metamorphism evolution laws but also various equations based on microstructure properties in other simulated processes. All modified equations are provided in the following section or in ~~Appendix~~an appendix. In addition, a historical tracker h_i is used as a last state variable with its meaning summarized in Table 2.

Light Absorbing Particles (LAP) are organic or mineral substances able to increase radiation absorption in snow. **?** added LAP mass contents $\mathcal{M}_{i,j}$ ($\mathrm{kg\,m^{-2}}$) as new optional state variables for each layer i and LAP type j. Although the code is designed to deal with any number of LAP types, only the parameters corresponding to black carbon and mineral dust are

Table 2. Possible values of the historical tracker h_i

	Liquid water at any time before t	Faceted crystals at any time before t	More than one freezing-melting cycle before t.
0	No	No	No
1	No	Yes	No
2	Yes	No	No
3	Yes	Yes	No
4	Yes	No	Yes
5	Yes	Yes	Yes

implemented in version 3.0.1. LAP interact with the other variable states of the model through absorption of solar radiation when the TARTES optical scheme is activated (Section 2.4.9).

Note that the layer thickness z_i (m), the layer temperature T_i (K) and the layer mass of liquid water l_i ($\mathrm{kg\,m^{-2}}$) can be directly derived from the state variables and are used in many parts of the model.

$$z_i = \frac{m_i}{\rho_i} \tag{2}$$

$$T_i = \min(T_0, T_i^*) \tag{3}$$

$$l_i = \max(0, T_i^* - T_0)\frac{m_i c_I}{L_m} \tag{4}$$

where $T_i^* = T_0 + \dfrac{H_i}{m_i c_I} + \dfrac{L_m}{c_I}$ (5)

T_0 (K) is the water ~~triple point temperature ,~~ melting point temperature which in SURFEX is assumed to be equal to the triple point value. L_m ($\mathrm{J\,kg^{-1}}$) is the latent heat of fusion, and c_I ($\mathrm{J\,kg^{-1}\,K^{-1}}$) the ice thermal capacity. Note also that the temperature θ_i in °C is also used in some parameterizations:

$$\theta_i = T_i - T_0 \tag{6}$$

2.3 Layering

2.3.1 Vertical discretization

One of the main original feature of Crocus compared to the majority of snowpack schemes available in the literature is its Lagrangian vertical discretization based on a dynamical evolution of the number and thicknesses of the numerical snow layers. The number of active layers N varies between $N_{\min} = 3$ and a user-defined maximum $N_{\max}$ (the default value is 50). Each layer is referenced by the index i which varies from 1 for the surface layer to N for the deepest layer. The main discretization principles described in **?** are still valid in the current version but they have been a bit complexified to solve numerical issues in specific applications. The current article provides the first comprehensive formulation of the algorithm.

First, a similarity criterion $D(i, i+1)$ between adjacent layers i and $i+1$ is defined. The initial formulation was a Manhattan distance of weighted values of dendricity, sphericity and grain size but it has never been published. The strict variable transformation of **?** led to a unjustified complexity in the formulation of this distance. Therefore, this distance was simplified in the same spirit by:

$$\begin{cases} D(i,i+1) & = 50\left(|S_{i+1}-S_i| + 2000\,|d_{i+1}-d_i| + \delta_h(i,i+1)\right) \quad \text{if } (A_i - A_{i+1}) < 300 \\ D(i,i+1) & = 200 \quad \text{otherwise} \end{cases} \tag{7}$$

where

$$\begin{cases} \delta_h(i,i+1) = 1 & \text{if } (h_i > 1 \text{ and } h_{i+1} \leq 1) \text{ or } (h_i \leq 1 \text{ and } h_{i+1} > 1) \\ \delta_h(i,i+1) = 0 & \text{otherwise} \end{cases} \tag{8}$$

The 2000 coefficient is chosen for normalization considering the typical range of 5.10^{-4}m between highest and lowest values of d_i (**?**) and the range of 1 between highest and lowest values of S_i. The δ_h function avoids the aggregation between cold snow and wet/refrozen snow. The specific case of significant age difference was introduced to avoid the aggregation of a recent snow layer with layers describing permanent snow or glacier. This distance is noted $D(n,1)$ when it is applied between a new snowfall and the surface layer.

When it is necessary to choose two layers to aggregate, a penalty criteria $P(i, i+1)$ is defined by:

$$P(i,i+1) = D(i,i+1) + 25\left(\frac{z_i}{z_i^*} + \frac{z_{i+1}}{z_{i+1}^*}\right) \tag{9}$$

where z_i^* (m) is the layer thickness of the optimal attractor profile defined in Appendix A.

Layers l_{sup} and l_{inf} are the adjacent layers which satisfy:

$$P(l_{\text{sup}}, l_{\text{inf}}) = \min_{i \in [1, N-1]} \left(P(i,i+1)\right) \tag{10}$$

The algorithm modifies the thicknesses at the previous time step $z_i^{t-\Delta t}$ for $i \in [1, N^{t-\Delta t}]$ towards a new profile z_i^t for $i \in [1, N^t]$ following ~~Equation~~ Eq. 11 by a succession of exclusive conditional statements depending on the total initial depth $Z^{t-\Delta t} = \sum_{i=1}^{N} z_i^{t-\Delta t}$, the initial thickness of the first layer $z_1^{t-\Delta t}$, the thickness of new snowfall to add $z_n = \frac{m_n}{\rho_n}$ (m), and the initial number of layers $N^{t-\Delta t}$. The computations of the mass m_n (kg m^{-2}) and density of new snowfall ρ_n (kg m^{-3}) are described later in Section 2.4.1.

$$
\begin{cases}
\text{If } z_n > 0 & \begin{cases}
\begin{bmatrix}\text{If } Z^{t-\Delta t} < Z_{\min} \\ \text{or } N^{t-\Delta t} < N_{\min}\end{bmatrix} & \begin{cases} N^t = \max\left[\mathrm{N_{min}}, \min(\mathrm{N_{max}}, \lfloor 100 \mathrm{z_n} \rfloor)\right] \\ z_i^t = \frac{z_n}{N^t} \forall i \in [1, N^t] \end{cases} \\
\text{else} & \begin{cases}
\begin{bmatrix} \text{If } z_1^{t-\Delta t} < z_1^* \text{ and } D(n,1) < 0.2 \\ \text{or } \dfrac{m_n}{\Delta t} < 3\times 10^{-5} \mathrm{kg\,m^{-2}s^{-1}} \text{ and } z_1 < 2 z_1^* \\ \text{or } z_1^{t-\Delta t} < 10^{-4} \end{bmatrix} & \begin{cases} N^t = N^{t-\Delta t} \\ z_1^t = z_1^{t-\Delta t} + z_n \\ z_i^t = z_i^{t-\Delta t} \forall i \in [2, N^t] \end{cases} \\
\text{else} & \begin{cases}
\text{If } N^{t-\Delta t} < N_{\max} & \begin{cases} N^t = N^{t-\Delta t} + 1 \\ z_1^t = z_n \\ z_i^t = z_{i-1}^{t-\Delta t} \forall i \in [2, N^t] \end{cases} \\
\text{else} & \begin{cases} N^t = N_{\max} \\ z_1^t = z_n \\ z_i^t = z_{i-1}^{t-\Delta t} \forall i \in [2, l_{\sup}] \\ z_{l_{\inf}}^t = z_{l_{\sup}} + z_{l_{\inf}} \\ z_i^t = z_i^{t-\Delta t} \forall i \in [l_{\inf}+1, N_{\max}] \end{cases}
\end{cases}
\end{cases}
\end{cases} \\
\text{else} & \begin{cases}
\text{If } z_1^{t-\Delta t} < 5.10^{-3} & \begin{cases} N^t = N^{t-\Delta t} - 1 \\ z_1^t = z_1^{t-\Delta t} + z_2^{t-\Delta t} \\ z_i^t = z_{i+1}^{t-\Delta t} \forall i \in [2, N^t] \end{cases} \\
\text{else} & \begin{cases}
\text{If } z_{N^{t-\Delta t}}^{t-\Delta t} < 10^{-2} & \begin{cases} N^t = N^{t-\Delta t} - 1 \\ z_i^t = z_i^{t-\Delta t} \forall i \in [1, N^t - 1] \\ z_{N^t}^t = z_{N^t}^{t-\Delta t} + z_{N^t+1}^{t-\Delta t} \end{cases} \\
\text{else} & \begin{cases}
\text{If } \exists j \in [1, \min(N^t - 1, N_{\max} - 4)] | z_j^{t-\Delta t} > \left(8 - \frac{N_{\max} - N^{t-\Delta t}}{N^{t-\Delta t} - N_{\min}}\right) z_j^* & \begin{cases} N^t = N^{t-\Delta t} + 1 \\ z_i^t = z_i^{t-\Delta t} \forall i \in [1, j-1] \\ z_i^t = \frac{z_j^{t-\Delta t}}{2} \forall i \in [j, j+1] \\ z_i^t = z_{i-1}^{t-\Delta t} \forall i \in [j+2, N^t] \end{cases} \\
\text{else} & \begin{cases}
\text{If } \exists j \in [2, N^t - 1] | \begin{bmatrix} z_j^{t-\Delta t} < \dfrac{5 z_j^*}{D(i, i+1)} \\ \text{and } \left(z_j^{t-\Delta t} + z_{j+1}^{t-\Delta t}\right) < 4.5 \max\left(z_j^*, z_{j+1}^*\right) \end{bmatrix} & \begin{cases} N^t = N^{t-\Delta t} - 1 \\ z_i^t = z_i^{t-\Delta t} \forall i \in [1, j-1] \\ z_j^t = z_j^{t-\Delta t} + z_{j+1}^{t-\Delta t} \\ z_i^t = z_{i+1}^{t-\Delta t} \forall i \in [j+1, N^t] \end{cases} \\
\text{else} & \begin{cases} N^t = N^{t-\Delta t} \\ z_i^t = z_i^{t-\Delta t} \forall i \in [1, N^t] \end{cases}
\end{cases}
\end{cases}
\end{cases}
\end{cases}
\end{cases}
\tag{11}
$$

In the definition of the initial number of layers (first line of Eq. 11), $\lfloor . \rfloor$ designates the floor operator. The literal translation of Eq. 11 is as follows. A uniform layering is applied for new snowfall on the ground while the total snow depth does not reach $Z_{\min} = 0.03$ m. In other cases, snowfall is aggregated to the surface layer when it is similar to a sufficiently thin surface layer or when it is very low. When new snowfall is too thick or too different from the surface layer, a new layer is created~~,~~ ~~optionally after the~~ . However, if $N_{\max}$ was already reached, this layer creation is done after the preliminary aggregation of the two closest layers of the profile $l_{\sup}$ and $l_{\inf}$ (as defined by ~~Equation 10), if $N_{\max}$ is reached~~Eq. 10). When there is no snowfall, the algorithm applies only one of the following modification by order of priority: aggregation of surface layer or bottom layer when too thin, split of internal layer when too thick, aggregation of internal layer when too thin, ~~relatively~~ relative to the optimal attractor profile. Note than only one modification by point is allowed at each time step. Thus, when the conditions for an internal split or aggregation are obtained for several layers (i.e. several values of j) at the same time, it is only applied for the uppermost layer $\min(j)$. For glacier applications, aggregation is strictly forbidden between snow and ice layers (i.e. when $\rho_i < \rho_G$ and $\rho_{i+1} > \rho_G$ where the threshold density for glacier ρ_G can be adjusted by the user (default $850\,\mathrm{kg\,m^{-3}}$).

This algorithm is applied independently at each simulation point. Therefore, the vertical layering differs between points in terms of number and thickness of snow layers. This raises a number of vectorization issues in the management of loops which were the topic of recent investigations detailed in Appendix **??**.

2.3.2 Aggregation of layers

When two layers i and $i+1$ are aggregated ($z_i^t = z_i^{t-\Delta t} + z_{i+1}^{t-\Delta t}$ from Eq. 11) the state variables are modified in order to conserve mass and enthalpy. The optical diameter is updated in order to obtain a mass-weighted average of the corresponding albedo. A mass-weighted average is also applied for sphericity and age.

$$m_i^t = m_i^{t-\Delta t} + m_{i+1}^{t-\Delta t} \tag{12}$$

$$\rho_i^t = \frac{m_i^t}{z_i^t} \tag{13}$$

$$H_i^t = H_i^{t-\Delta t} + H_{i+1}^{t-\Delta t} \tag{14}$$

$$d_i^t = \left[\frac{0.9 - \frac{m_i^{t-\Delta t}\left(0.9 - 15.4\sqrt{d_i^{t-\Delta t}}\right) + m_{i+1}^{t-\Delta t}\left(0.9 - 15.4\sqrt{d_{i+1}^{t-\Delta t}}\right)}{m_i + m_{i+1}}}{15.4} \right]^2 \tag{15}$$

$$S_i^t = \frac{m_i^{t-\Delta t} S_i^{t-\Delta t} + m_{i+1}^{t-\Delta t} S_{i+1}^{t-\Delta t}}{m_i + m_{i+1}} \tag{16}$$

$$A_i^t = \frac{m_i^{t-\Delta t} A_i^{t-\Delta t} + m_{i+1}^{t-\Delta t} A_{i+1}^{t-\Delta t}}{m_i + m_{i+1}} \tag{17}$$

$$h_i^t = \begin{cases} h_i^{t-\Delta t} \text{ if } m_i^{t-\Delta t} \geq m_{i+1}^{t-\Delta t} \\ h_{i+1}^{t-\Delta t} \text{ otherwise} \end{cases} \tag{18}$$

$$\mathcal{W}_{d,i}^t = \mathcal{W}_{d,i}^{t-\Delta t} + \mathcal{W}_{d,i+1}^{t-\Delta t} \tag{19}$$

$$\mathcal{W}_{w,i}^t = \mathcal{W}_{w,i}^{t-\Delta t} + \mathcal{W}_{w,i+1}^{t-\Delta t} \tag{20}$$

The same equations apply to aggregate falling snow with the surface layer replacing x_i^t, $x_i^{t-\Delta t}$ and $x_{i+1}^{t-\Delta t}$ by respectively x_1^t, $x_1^{t-\Delta t}$ and x_n for each x representing all state variables m, ρ, H, d, S, A, h, $\mathcal{W}_d$ and $\mathcal{W}_w$ in Eq. 12-20.

2.3.3 Splitting of layers

When a layer i is ~~splitted~~ split into two layers i and $i+1$ ($z_i^t = z_{i+1}^t = \frac{z_i^{t-\Delta t}}{2}$ from Eq. 11), mass and enthalpy are divided into equal parts and microstructure properties are not modified:

$$m_i^t = m_{i+1}^t = \frac{m_i^{t-\Delta t}}{2} \tag{21}$$

$$\rho_i^t = \frac{m_i^t}{z_i^t} \tag{22}$$

$$H_i^t = H_{i+1}^t = \frac{H_i^{t-\Delta t}}{2} \tag{23}$$

$$d_i^t = d_{i+1}^t = d_i^{t-\Delta t} \tag{24}$$

$$S_i^t = S_{i+1}^t = S_i^{t-\Delta t} \tag{25}$$

$$A_i^t = A_{i+1}^t = A_i^{t-\Delta t} \tag{26}$$

$$h_i^t = h_{i+1}^t = h_i^{t-\Delta t} \tag{27}$$

$$\mathcal{W}_{d,i}^t = \mathcal{W}_{d,i+1}^t = \frac{\mathcal{W}_{w,i}^{t-\Delta t}}{2} \tag{28}$$

$$\mathcal{W}_{w,i}^t = \mathcal{W}_{w,i+1}^t = \frac{\mathcal{W}_{w,i}^{t-\Delta t}}{2} \tag{29}$$

(30)

2.4 Evolution equations

2.4.1 Snowfall

The new snow amount is the result of solid precipitation, but also deposition of blowing snow and snowmaking, when the corresponding modules are activated. These three mass sources are respectively refered by the subscripts SP, BS and SM in the following. In case of simultaneous occurrence, only one additional layer is created with weighted physical properties. Thus, the mass of the new layer is defined by:

$$m_n = m_{\mathrm{SP}} + m_{\mathrm{BS}} + m_{\mathrm{SM}} \tag{31}$$

The density of new snow ρ_n, optical diameter d_n, and sphericity S_n are expressed by:

$$\rho_n = \frac{\rho_{\mathrm{SP}} m_{\mathrm{SP}} + \rho_{\mathrm{BS}} m_{\mathrm{BS}} + \rho_{\mathrm{SM}} m_{\mathrm{SM}}}{m_n} \tag{32}$$

$$d_n = \frac{d_{\mathrm{SP}} m_{\mathrm{SP}} + d_{\mathrm{BS}} m_{\mathrm{BS}} + d_{\mathrm{SM}} m_{\mathrm{SM}}}{m_n} \tag{33}$$

$$S_n = \frac{S_{\mathrm{SP}} m_{\mathrm{SP}} + S_{\mathrm{BS}} m_{\mathrm{BS}} + S_{\mathrm{SM}} m_{\mathrm{SM}}}{m_n} \tag{34}$$

The mass of solid precipitation during the time step Δt is directly provided by the forcing:

$$m_{\mathrm{SP}} = \mathcal{P}_s \Delta t \tag{35}$$

Several empirical expressions of the density of natural falling snow ρ_{SP} are implemented (**???**), as a function of 2-m air temperature T_a^* (°C) $= T_a - T_0$ and 10-m wind speed U_{10} $(\mathrm{m\,s^{-1}})$. They can be activated through the SNOWFALL physical option following ~~Equation~~ Eq. 36. An illustration comparing these formulations is provided in **?**, Figure 1 in the corrigendum. The parameters are listed in Appendix **??**. When the forcing wind speed is not available at a 10 m height, a logarithmic adjustment is applied following Appendix B.

$$\begin{cases} \text{If SNOWFALL = V12:} & \rho_{\mathrm{SP}} = \max(\rho_{\min}, a_\rho + b_\rho T_a^* + c_\rho \sqrt{U_{10}}) \\ \text{If SNOWFALL = S14:} & \begin{cases} \log_{10}(\rho_{\mathrm{SP}}) = e_\rho + f_\rho T_a^* + g_\rho + h_\rho \sin^{-1}(\sqrt{i_\rho}) + j_\rho \log_{10}(\max[U_{10}, 2]) & \text{if } T_a^* \geq -14^oC \\ \log_{10}(\rho_{\mathrm{SP}}) = e_\rho + f_\rho T_a^* + h_\rho \sin^{-1}(\sqrt{i_\rho}) + j_\rho \log_{10}(\max[U_{10}, 2]) & \text{otherwise} \end{cases} \\ \text{If SNOWFALL = A76:} & \rho_{\mathrm{SP}} = \rho_{\min} + \max(k_\rho (T_a^* + l_\rho)^{1.5}, 0) \end{cases} \tag{36}$$

Adjustments of parameter c_ρ were proposed by **?** to account for the uncertainties associated with the impact of wind speed on Arctic snowfall density. Although hard-coded in version 3.0.1, this parameter will be tunable in a future version.

Several formulations of microstructure properties of new snow were implemented by **?** depending on SNOWDRIFT option. Again, Appendix B is used to assess the 5-m wind speed U_5 from the forcing wind speed U at heigth z_u. SNOWDRIFT options are reformulated here with the new microstructure variables now used in the model:

$$\begin{cases} \text{If SNOWDRIFT=NONE} \begin{cases} S_{\mathrm{SP}} = \min\left(\max\left(0.0795 U_5 + 0.38, 0.5\right), 0.9\right) \\ d_{\mathrm{SP}} = 10^{-4}\left(\delta_{\mathrm{NONE}} + (1 - \delta_{\mathrm{NONE}})(4 - S_{\mathrm{SP}})\right) \\ \text{where } \delta_{\mathrm{NONE}} = \min\left(\max(1.29 - 0.173 U_5, 0.2), 1\right) \end{cases} \\ \text{If SNOWDRIFT=DFLT} \begin{cases} S_{\mathrm{SP}} = 0.5 \\ d_{\mathrm{SP}} = 10^{-4} \end{cases} \\ \text{If SNOWDRIFT=VI13} \begin{cases} S_{\mathrm{SP}} = \min\left(\max\left(0.035 U_5 + 0.43, 0.5\right), 0.9\right) \\ d_{\mathrm{SP}} = 10^{-4}\left(\delta_{\mathrm{VI13}} + (1 - \delta_{\mathrm{VI13}})(4 - S_{\mathrm{SP}})\right) \\ \text{where } \delta_{\mathrm{VI13}} = \min\left(\max(1.14 - 0.07 U_5, 0.2), 1\right) \end{cases} \\ \text{If SNOWDRIFT=GA01} \begin{cases} S_{\mathrm{SP}} = 1 - 0.49 \delta_{\mathrm{GA01}} \\ d_{\mathrm{SP}} = 10^{-4}\left(\delta_{\mathrm{GA01}} + (1 - \delta_{\mathrm{GA01}})(4 - S_{\mathrm{SP}})\right) \\ \text{where } \delta_{\mathrm{GA01}} = \min(\max(2.868 \exp(-0.085 U_5) - 1, 0), 1) \end{cases} \end{cases} \tag{37}$$

VI13 refers to the approach of **?** and GA01 refers to the approach of **?**. The initial value of the historical tracker is $h_n = 0$ and the initial value of snow age is $A_n = 0$. At each following time step, this variable remembers the duration from the snowfall event by the simple evolution:

$$A_i^{t+\Delta t} = A_i^t + \frac{\Delta t}{86400} \tag{38}$$

2.4.2 Blowing snow

The mass, density, optical diameter and sphericity of blowing snow (m_{BS}, ρ_{BS}, d_{BS}, S_{BS}) can be provided by an external snow transport module. Within SURFEX, three different snow transport modules can be coupled to Crocus. The SYTRON module (**??**) is designed to simulate erosion and accumulation from the windward to the leeward side of the crests in French operational simulations based on an idealized topography. The SnowPappus module comprehensively described by **?** is designed for gridded simulations at horizontal resolutions between 30 and 250 m (**?**). Finally, snow transport can be solved by an explicit coupling with the MESO-NH Large Eddy Simulation model for process studies on small study domains and at the event scale (**??**).

2.4.3 Machine-made snow

A representation of machine-made snow can optionally be provided by a dedicated module activated by the logical option SNOWMAK_BOOL. This enables the model to compute the mass, density, optical diameter and sphericity of machine-made snow (snowmaking) (m_{SM}, ρ_{SM}, d_{SM}, S_{SM}) and its interaction with the rest of the snowpack. First, the wet bulb temperature T_w (expressed in K) is computed following Eq. F20 in Appendix F. The machine-made production is allowed if both the wet bulb temperature and the wind speed are lower than the respective user-defined thresholds T_{lim} and U_{lim}. When the meteorological conditions are satisfied, snowmaking is activated only during two periods over the winter season: the "base-layer generation" production period and the "reinforcement" production period (**??**). The dates of beginning and end of each period and the times allowed in the day are defined by the user from parameters day_1 to day_3 and t_1 to t_4 described in Appendix **??**. Two different strategies can be adopted depending on the SELF_PROD logical option. When it is set to True, the production follows a pre-defined set of rules. In this case, during the base-layer generation period the production is allowed until a given amount of water p_{lim} ($\mathrm{kg\ m^{-2}}$) is used. During the reinforcement period, instead, the production is allowed if the total snow depth Z is lower than a threshold Z_{lim}. If SELF_PROD is False, the production is forced to match a water use target ($\mathrm{kg\ m^{-2}}$) defined by the user for each simulated point, regardless of any meteorological or timing condition.

The mass of machine-made snow is then obtained by:

$$m_{\mathrm{SM}} = (1 - \mathcal{L}_{\mathrm{SM}}) \frac{\mathcal{I}_{\mathrm{SM}}}{\mathcal{A}_{\mathrm{SM}}} \left(a_{\mathrm{SM}}(T_w - T_0) + b_{\mathrm{SM}}\right) \Delta t \tag{39}$$

where $\mathcal{I}_{\mathrm{SM}} = 1$ when production is allowed, and 0 otherwise, $\mathcal{A}_{\mathrm{SM}}$ is the surface area covered by a snow gun set to 3300 $\mathrm{m^2}$, and $\mathcal{L}_{\mathrm{SM}}$ the loss factor set to 0.4 by **?**. a_{SM} ($\mathrm{kg\ K^{-1}\ s^{-1}}$) and b_{SM} ($\mathrm{kg\ s^{-1}}$) are regression coefficients of the parameterization of the potential mass produced by a snowgun. They can be adjusted by the user or derivated from **?**.

The density ρ_{SM} of machine-made snow can be adjusted by the user whereas the snow microstructure properties d_{SM} and S_{SM} are fixed following **?**.

For more information about the implementation of the snowmaking practices in Crocus, please refer to **?** and **?**.

2.4.4 Freezing rain

When liquid precipitation occurs at negative temperatures, the model can simulate the formation of an ice layer at the top of the surface following the work of **?**. In a first step, the whole precipitation mass is assumed to form a new ice layer at T_0, only aggregated to the previous surface layer if a thin ice layer is already present (Eq. 40)

$$\text{If } \mathcal{P}_r > 0 \text{ and } T_a < T_0 : \begin{cases} m_n = \mathcal{P}_r \Delta t \\ \rho_n = \rho_I \\ H_n = -L_m \mathcal{P}_r \Delta t \\ A_n = 0 \\ d_n = 2 \times 10^{-3} \mathrm{m} \\ S_n = 1 \\ h_n = 2 \\ \text{If } (D(n,1) < 20 \text{ and } z_1^- < 0.01) \text{ or } N^- = N_{\max} : \begin{cases} N^+ = N^- \\ \text{Eq. } 12-20 \text{ applied to } (x_n, x_1^-) \\ \forall x \in [m,\rho,H,A,d,S,h] \end{cases} \\ \text{else : } \begin{cases} N^+ = N^- + 1 \\ x_1^+ = x_n \forall x \in [m,\rho,H,A,d,S,h] \\ x_i^+ = x_{i-1}^- \forall x \in [m,\rho,H,A,d,S,h] \text{ and } \forall i \in [2, N^+] \end{cases} \end{cases} \tag{40}$$

The energy associated with the phase change E_{FRZ} ($\mathrm{W\,m^{-2}}$) is computed by Eq. 41 and accounted for further as an additional energy source in the surface energy balance (Eq. 94), able to partly melt this new ice layer:

$$\begin{cases} \text{If } \mathcal{P}_r > 0 \text{ and } T_a < T_0 : & E_{\mathrm{FRZ}} = (1 - \phi_{\mathrm{FRZ}}) L_m \mathcal{P}_r \Delta t \text{ where } \phi_{\mathrm{FRZ}} = \frac{c_W}{L_m}(T_0 - T_a) \\ \text{else :} & E_{\mathrm{FRZ}} = 0 \end{cases} \tag{41}$$

The assumption behind Eq.41 is that a fraction ϕ_{FRZ} of the latent heat release due to refreezing is consumed by the increase of temperature from T_a to T_0 and the remaining part is fully stored by the surface layer and can either be available for melting ~~either~~ or be partly dissipated through diffusion or heat exchanges with the atmosphere, as solved later in Sections 2.4.12 and 2.4.13.

Note that some other implementations of this process in other models consider than while all the precipitation eventually freezes and contributes to the formation of an ice layer at the surface, only a fraction of the associated latent heat is kept in the ice layer, while the remaining latent heat is transferred to the atmosphere at a shorter time scale than the model time step (**??**).

2.4.5 Metamorphism

The prognostic equations of microstructure variables d_i and S_i are still fully empirical in the absence of physical evolution laws. They depend on conditions ~~on~~ of the vertical temperature gradient $\mathcal{G}_i$ estimated by:

$$\begin{cases} \mathcal{G}_1 = \frac{2|T_2 - T_1|}{z_1 + z_2} \\ \mathcal{G}_i = \frac{2|T_{i+1} - T_{i-1}|}{z_{i-1} + 2z_i + z_{i+1}} \forall i \in [2, N-1] \\ \mathcal{G}_N = \frac{2|T_N - T_{N-1}|}{z_{N-1} + z_N} \end{cases} \tag{42}$$

Following **?** and **?**, the evolution of sphericity ΔS_i during a time step Δ_t is defined by Eq. 43 (dry metamorphism) and 44 (wet metamorphism):

$$\text{If } l_i = 0: \begin{cases} \text{If } \mathcal{G}_i < 5\,\mathrm{K\,m^{-1}} & \begin{cases} \text{If } S_i < 1 & \Delta S_i = \mathrm{sph}_1 e^{-6000/T_i} f_{\text{corr}}(d_i, S_i, h_i)\Delta t \\ \text{else} & \Delta S_i = 0 \end{cases} \\ \text{else} & \begin{cases} \text{If } S_i > 0 & \Delta S_i = -\mathrm{sph}_2 e^{-6000/T_i} \mathcal{G}_i^{0.4}\Delta t \\ \text{else} & \Delta S_i = 0 \end{cases} \end{cases} \tag{43}$$

$$\text{If } l_i > 0: \begin{cases} \text{If } S_i < 1: \quad \Delta S_i = \max\left(\mathrm{sph}_3\left(100\frac{l_i}{\rho_i z_i}\right)^3, \mathrm{sph}_2 e^{-6000/T_0}\right)\Delta t \\ \text{If } S_i = 1: \quad \Delta S_i = 0 \end{cases} \tag{44}$$

Parameters sph_1, sph_2 and sph_3 are provided in Appendix **??**. $f_{\text{corr}}(d_i, S_i, h_i)$ is an unpublished parameterization in the original code of **?** which modifies the general behaviour of Eq. 43 by reducing the sphericity increase of snow layers with microstructure properties typical of depth hoar or large faceted crystals and submitted to low thermal gradients. This prevents the formation of rounded grains from depth hoar or large faceted crystals. Indeed, the persistence of anisotropy in this case was obtained with a phase-field numerical model applied to snow microstructure (**?**) and recently confirmed by unpublished tomography observations.

$$\begin{cases} \text{If } S_i \geq 0.5\,;\, h_i = 1 \text{ and } g_{s_i}(d_i, S_i) > 5\times10^{-4}\mathrm{m} & f_{\text{corr}}(d_i, S_i, h_i) = 0 \\ \text{If } S_i < 0.5\,;\, h_i = 1 \text{ and } g_{s_i}(d_i, S_i) > 5\times10^{-4}\mathrm{m} & f_{\text{corr}}(d_i, S_i, h_i) = \exp(\frac{3\times10^{-4} - g_{s_i}(d_i, S_i)}{10^{-4}}) \\ \text{otherwise} & f_{\text{corr}}(d_i, S_i, h_i) = 1 \end{cases} \tag{45}$$

h_i is a tracker of the snow layer history (Table 2) for which ~~the~~ odd values (1~~value~~ , 3, 5) corresponds to the occurrence of depth hoar at any time since the layer creation. $g_{s_i}(d_i, S_i)$ is a variable originally used to describe snow microstructure by **?** (grain size). Its retrieval from the current state variables d_i and S_i is described in Appendix D. It must be noticed that this parameterization was forgotten by **?** because it was unpublished and was only recently restored in the code.

Several prognostic evolutions of d_i during dry metamorphism were implemented by **?** and **?**, following the works from **?**, **?** and **?**. Furthermore, several issues were found by **?** in the translation of the **?** formalism (using dendricity and grain size) in terms of optical diameter evolution by **?** (details in Appendix D). A new parameterization (B21) was therefore recently implemented to solve the associated issues and now replaces the original implementation of **?** now removed from the code. The resulting available evolution laws of d_i during dry metamorphism are given by Eq. 46-49.

$$\begin{bmatrix}\text{If SNOWMETAMO=B21}\\ \text{and } l_i = 0:\end{bmatrix}\begin{cases}\text{If } d_i \geq 10^{-4}(4-S_i): & \begin{cases}\text{If } \mathcal{G}_i > 15\,\mathrm{K\,m^{-1}} \text{ and } S_i = 0: & \Delta d_i = \frac{1}{2} f(\theta_i) h(\rho_i) g(\mathcal{G}_i)\Phi\Delta t\\ \text{else:} & \Delta d_i = \frac{d_i - 4\times 10^{-4}}{1+S_i}\Delta S_i\end{cases}\\ \text{else :} & \Delta d_i = 10^{-4}\left[-\text{sph}_2 e^{-6000/T_i}\mathcal{J}_i\left(S_i-3\right) + \frac{\Delta S_i}{\Delta t}\frac{10^4 d_i - 1}{S_i - 3}\right]\Delta t\\ & \text{where}\begin{cases}\mathcal{J}_i = \mathcal{G}_i^{0.4} & \text{if } \mathcal{G}_i \geq 5\\ \mathcal{J}_i = 1 & \text{if } \mathcal{G}_i < 5\end{cases}\end{cases} \tag{46}$$

$$\begin{bmatrix}\text{If SNOWMETAMO=F06}\\ \text{and } l_i = 0:\end{bmatrix} \quad \Delta d_i = 2\dot{r}_0(\rho_i, T_i, \mathcal{G}_i)\left[\frac{\tau(\rho_i, T_i, \mathcal{G}_i)}{\frac{d_i}{2} - r_0 + \tau(\rho_i, T_i, \mathcal{G}_i)}\right]^{\frac{1}{\kappa(\rho_i, T_i, \mathcal{G}_i)}}\Delta t \tag{47}$$

$$\begin{bmatrix}\text{If SNOWMETAMO=S-B}\\ \text{and } l_i = 0:\end{bmatrix}\begin{cases}\text{If } A_i \leq 2\text{ days} & \Delta d_i = \frac{a_s \rho_i d_i^2}{6} + \frac{b_s T_i\left(\frac{6}{\rho_i}\right)^{m_s - 1}}{d_i^{m_s - 2}}\\ \text{else:} & \text{Eq. 46}\end{cases} \tag{48}$$

$$\begin{bmatrix}\text{If SNOWMETAMO=S-F}\\ \text{and } l_i = 0:\end{bmatrix}\begin{cases}\text{If } A_i \leq 2\text{ days} & \text{Eq. 48}\\ \text{else:} & \text{Eq. 47}\end{cases} \tag{49}$$

$$\tag{50}$$

Functions $f(\theta_i)$, $h(\rho_i)$, $g(\mathcal{G}_i)$ and parameter Φ used in Eq. 46 for the growth of faceted crystals in the case of high gradient metamorphism follow **?** and are defined in Appendix C. In Eq. 47 from **?**, coefficients $\dot{r}_0(\rho_i, T_i, \mathcal{G}_i)$, $\tau(\rho_i, T_i, \mathcal{G}_i)$ and $\kappa(\rho_i, T_i, \mathcal{G}_i)$ are retrieved from look-up tables provided in a parameters file (cf. section on Data and Code availability) and $r_o = 5\times 10^{-5}$ m. Experimental parameters a_s, b_s, m_s in Eq. 49 are defined in Appendix **??** from **?**.

Regardless of the SNOWMETAMO option, wet metamorphism is always parameterized with the laws published by **?**, rewritten here with the new microstructure prognostic variables and the methodology described in Appendix D:

$$\text{If } l_i > 0\begin{cases}\text{If } d_i < 10^{-4}(4-S_i): \begin{cases}\text{If } S_i < 1: \Delta d_i = 10^{-4}\left(\frac{10^4 d_i - 1}{S_i - 3} - (S_i - 3)\right)\Delta S_i\\ \text{If } S_i = 1: \Delta d_i = 2.10^{-4}\max\left(\text{sph}_3\left(100\frac{l_i}{\rho_i z_i}\right)^3, \text{sph}_2 e^{-6000/T_0}\right)\Delta t\end{cases}\\ \text{else :}\begin{cases}\text{If } S_i < 1: \Delta d_i = \frac{d_i - 4.10^{-4}}{S_i + 1}\Delta S_i\\ \text{If } S_i = 1: \Delta d_i = \frac{2}{\pi}\frac{\left[v_0 + v_1\left(100\frac{l_i}{\rho_i z_i}\right)^3\right]}{d_i^2}\Delta t\end{cases}\end{cases} \tag{51}$$

To follow its definition in Table 2, the historical tracker h_i is updated at the end of this subroutine by:

$$
\begin{cases}
\text{If } d_i \geq 10^{-4}(4-S_i): \begin{cases}
\text{If } S_i = 0 \text{ and } h_i^t = 0 : h_i^{t+\Delta t} = 1 \\
\text{else}: \begin{cases}
\text{If } S_i = 1 \text{ and } \frac{l_i}{\rho_i z_i} > 0.005 : \begin{cases}
\text{If } h_i^t = 0 : h_i^{t+\Delta t} = 2 \\
\text{If } h_i^t = 1 : h_i^{t+\Delta t} = 3 \\
\text{else}: h_i^{t+\Delta t} = h_i^t
\end{cases} \\
\text{else}: \begin{cases}
\text{If } T_i < T_0 : \begin{cases}
\text{If } h_i^t = 2 : h_i^{t+\Delta t} = 4 \\
\text{If } h_i^t = 3 : h_i^{t+\Delta t} = 5 \\
\text{else}: h_i^{t+\Delta t} = h_i^t
\end{cases} \\
\text{else}: h_i^{t+\Delta t} = h_i^t
\end{cases}
\end{cases}
\end{cases} \\
\text{else } h_i^{t+\Delta t} = h_i^t
\end{cases}
\tag{52}
$$

2.4.6 Natural compaction

For a given layer of density ρ_i the mechanical settling under the over burden σ_i (Pa) is expressed with a visco-elastic model (**??**):

$$
\text{If SNOWCOMP} \in [\text{B92, T11}] : \Delta\rho_i = \frac{\rho_i \sigma_i}{\eta_i} \Delta t \tag{53}
$$

where

$$
\begin{cases}
\sigma_i = g \cos\Theta \sum_{j=1}^{i-1} \rho_j z_j \forall i \in [2, N] \\
\sigma_1 = g \cos\Theta \times 0.5 \rho_1 z_1
\end{cases}
\tag{54}
$$

g is the gravitational acceleration and ~~γ~~ Θ the slope angle from horizontal.

The viscosity η_i is a function of density ρ_i and temperature θ_i ($^\circ C$) depending on the SNOWCOMP option (**??**):

$$
\begin{cases}
\text{If SNOWCOMP = B92:} \quad \eta_i = f_1(w_i) f_2(d_i, S_i) \eta_0 \frac{\rho_i}{c_\eta} e^{a_\eta(-\theta_i) + b_\eta \rho_i} \\
\text{If SNOWCOMP = T11:} \quad \eta_i = f_1(w_i) f_2(d_i, S_i) * 0.05 \rho_i^{-0.0371\theta_i + 4.4} (10^{-4} e^{0.018\rho_i} + 1)
\end{cases}
\tag{55}
$$

The parameters for case B92 are defined in Appendix **??**. The multiplicative function f_1 accelerates the settlement of wet snow as a function of the volumetric liquid water content w_i. The multiplicative function f_2 reduces the settlement of layers consisting of faceted snow types as a function of microstructure properties d_i and S_i (translation of the formula from **?** in the new formalism).

$$f_1(w_i) = \frac{1}{1 + 60\frac{w_i}{\rho_w}} \tag{56}$$

$$f_2(d_i, S_i) = \begin{cases} \min\left(4, \exp\left(\min\left(4, g_{s_i}(d_i, S_i) \times 10^4 - 2\right)\right)\right) & \text{if } d_i > (4 - S_i) \times 10^{-4} \text{ and } S_i \leq 0.5 \\ 1 & \text{otherwise} \end{cases} \tag{57}$$

where $g_{s_i}(d_i, S_i)$ is defined by Eq. D6. The compaction rate however has a complex dependence ~~to~~ on snow microstructure (**?**) which cannot be described by the representation of snow microstructure in Crocus. Alternatively to ~~equation~~ Eq. 53, it is possible to use a parameterization from **?** derived from tomographic observations and representing a non-linear relationship between settlement, the stress σ_i (Pa) and the optical diameter increase for the first 48 hours after snowfall:

$$\text{If SNOWCOMP = S14:} \begin{cases} \Delta\rho_i = B_S \frac{\Delta d_i}{d_i^2} \sigma_i^{k_S} & \text{if } A_i \leq 2 \text{ days} \\ \text{Eq. 53} & \text{if } A_i > 2 \text{ days} \end{cases} \tag{58}$$

with $B_S = 3.96 \times 10^{-2}$ and $k_S = 0.18$. The current Crocus parameterization is applied when the snow layer age exceeds 2 days.

Alternative parameterizations reducing compaction in the presence of low vegetation were proposed by **?** combining earlier works from **?** and **?**. Although not available in version 3.0.1, they will be implemented in a future version.

2.4.7 Grooming

The effects of grooming machines (snowcats) on the snow physical properties include the compaction induced by their overburden weight and the mixing of surface layers produced by the tiller. Both these effects can optionally be simulated in Crocus (**?**).

The compaction effect is activated using the logical switch SNOWCOMPACT and only applies if the total mass $M = \sum_{i=1}^{N} m_i$ is higher than 20 $\mathrm{kg\ m^{-2}}$ (a minimum value required by the grooming machines) and between 8pm and 9pm (and also 6am-9am in case of snowfall during the night). Grooming starts on November 1st and continues until the date $\mathrm{day_{END}}$ chosen by the user. Its frequency f_{GRO} (number of grooming sessions per day) is also user-defined. It is worth noticing that it is possible to activate grooming without activating snowmaking, but the opposite would not be realistic. The static stress due to the weight of the snowcat itself σ_{GRO_i} (Pa) is expressed for layer i by:

$$\text{If SNOWCOMPACT} \begin{cases} \text{If} \quad \sum_{j=1}^{i} m_j < 50\ \mathrm{kg\ m^{-2}} : \sigma_{\mathrm{GRO}_i} = g\cos\Theta \times 500 \\ \text{If} \quad 50\ \mathrm{kg\ m^{-2}} \leq \sum_{j=1}^{i} m_j < 150\ \mathrm{kg\ m^{-2}} : \sigma_{\mathrm{GRO}_i} = g\cos\Theta \times (150 - \sum_{j=1}^{i} m_j) * 5 \\ \text{If} \quad \sum_{j=1}^{i} m_j \geq 150\ \mathrm{kg\ m^{-2}} : \sigma_{\mathrm{GRO}_i} = 0 \end{cases} \tag{59}$$

and the corresponding density change is obtained by replacing σ_i by σ_{GRO_i} in Eq. 53.

The tilling effect is activated using the logical switch SNOWTILLER, applies down to 35 $\mathrm{kg\,m^{-2}}$ below the surface and only if SNOWCOMPACT = True. The tiller mounted at the rear of snowcats produces two main effects: it further increases the density of the snow by loading the snowpack with extra pressure and it modifies the snow microstructure by creating smaller, rounded grains. As a result, all impacted layers are mixed together, their properties are homogenized and some of them are modified (**?**). These effects are simulated in Crocus by directly modifying the density, optical diameter and sphericity of the impacted layers. The density reached by the snowpack after grooming is parameterized as:

$$\text{If SNOWTILLER: } \rho_i = \max\left(\overline{\rho}, \frac{2\overline{\rho} + 3\rho_{\mathrm{GRO}}}{5}\right) \forall i \in [1,k] | \sum_{i=1}^{k} m_i <= 35 \mathrm{kg\ m^{-2}} \tag{60}$$

where $\overline{\rho} = \frac{\sum_{i=1}^{k}\rho_i m_i}{\sum_{i=1}^{k} m_i}$ is the weighted average density of the k impacted layers and ρ_{GRO} is the target density that should eventually be reached by the grooming process (**??**). The optical diameter and sphericity of snow are altered analogously, using the respective target values d_{GRO} and S_{GRO} (Appendix **??**, **??**).

2.4.8 Drifting snow

Even if the Crocus model is not coupled with a dedicated module able to transport snow from one simulation point to another, it is possible to activate a parameterization from **?** simulating the impact of drifting snow on compaction and metamorphism, but with mass conservation. This parameterization relies on two possible definitions of a mobility ~~indice~~ index MOB_i (**??**):

$$\begin{cases} \text{If SNOWMOB=GM98} & \begin{cases} \text{If } d_i < 10^{-4}(4 - S_i): & \mathrm{MOB}_i = 0.5(1 - S_i) + 0.75\delta_i \\ \text{else:} & \mathrm{MOB}_i = 0.833(1 - S_i) - 0.583 g_{s_i} \end{cases} \\ \text{If SNOWMOB=VI12} & \begin{cases} \text{If } d_i < 10^{-4}(4 - S_i): & \mathrm{MOB}_i = 0.34 \times (0.5(1 - S_i) + 0.75\delta_i) + 0.66F(\rho_i) \\ \text{else:} & \mathrm{MOB}_i = 0.34 \times (0.833(1 - S_i) - 0.583 g_{s_i}) + 0.66F(\rho_i) \end{cases} \end{cases} \tag{61}$$

$$\text{where } F(\rho) = [1.25 - 0.0042\,(\max(\rho_{\min}, \rho) - \rho_{\min})] \text{ and } \rho_{\min} = 50\ \mathrm{kg\,m^{-3}}. \tag{62}$$

In addition, a threshold is applied on the mobility ~~indice in case of melt forms~~index in case liquid water had been present at any time in the snow layer:

$$\text{If } h_i >= 2: \mathrm{MOB}_i = \min(-0.0583, \mathrm{MOB}_i \text{ from Eq.61}) \tag{63}$$

The mobility ~~indice~~ index was expressed with the original formalism of metamorphism. The conversion functions $\delta_i(d_i, S_i)$ and $g_{s_i}(d_i, S_i)$ are defined in Appendix D by Eq. D2 and D6. Redefining this ~~indice~~ index as a function of the current state variables (d_i and S_i) would be more consistent and flexible but it would require a significant new effort of evaluation. It highlights the issues involved by changes in the variable states in a model which have been underestimated in the work of **?**.

A driftability ~~indice~~ index $\mathcal{D}_i$ is then obtained by combining the mobility ~~indice~~ index and the 5 m wind speed U_5:

$$\mathcal{D}_i = \max\left(\mathrm{MOB}_i - (2.868 * \exp(-0.085 \times U_5) - 1.), 0\right) \tag{64}$$

We then introduce:

$$\mathcal{D}_{\mathrm{EFF}_i} = \mathcal{D}_i \times \exp\left(-10 \times \left(\sum_{j=1}^{i-1} z_j (3.25 - \mathcal{D}_j) + \frac{z_i}{2}(3.25 - \mathcal{D}_i)\right)\right) \tag{65}$$

to formulate the variation of density due to snow drift by:

$$\Delta\rho_i = \mathcal{D}_{\mathrm{EFF}_i} \max\left(\rho_{\mathrm{MAX}} - \rho_i, 0)\right) \frac{\Delta t}{\tau_{\mathrm{DRIFT}}} \tag{66}$$

with $\rho_{\mathrm{MAX}} = 350\ \mathrm{kg\ m^{-3}}$ and $\tau_{\mathrm{DRIFT}} = 172800$ s (2 days). Adjustments of parameters ρ_{MAX} and τ_{DRIFT} were proposed by several authors for Arctic snow as summarized by **?**. Although hard-coded in version 3.0.1, they will be adjustable by the user in a future version. Then, the variation of microstructure properties due to fragmentation during snow transport are obtained by Eq. 67. The origin of this expression is provided in Appendix E.

$$\begin{cases} \text{If } d_i < 10^{-4}(4 - S_i): & \Delta d_i = \mathcal{D}_{\mathrm{EFF}_i} \times 10^{-4} \left[(2.5 - 1.5 S_i)\delta_i - 1 + S_i\right] \frac{\Delta t}{\tau_{\mathrm{DRIFT}}} \\ \text{else} & \Delta d_i = \mathcal{D}_{\mathrm{EFF}_i} \left(-5 \times 10^{-4} \frac{1+S_i}{2} + (d_i - 4 \times 10^{-4}) \frac{1-S_i}{1+S_i}\right) \frac{\Delta t}{\tau_{\mathrm{DRIFT}}} \end{cases} \tag{67}$$

$$\Delta S_i = \mathcal{D}_{\mathrm{EFF}_i} (1 - S_i) \frac{\Delta t}{\tau_{\mathrm{DRIFT}}} \tag{68}$$

Optionally, a mass loss due to blowing snow sublimation can be estimated and removed from the surface layer. The parameterization is inspired from **?** with a modification of a threshold wind speed U_t to account for the microstructure-related mobility ~~indice~~index. This option is not activated by default due to the large associated uncertainties and lack of evaluation but is considered to be necessary in polar environments (**??**). The mass reduction of the surface layer due to sublimation Δm_1 is obtained by:

$$\Delta m_1 = \min\left(0.5 m_1, a_{\mathrm{SUBL}} \left(\frac{T_0}{T_a}\right)^{\gamma_{\mathrm{SUBL}}} U_t \rho_a q_{sat}(T_a) \left(1 - \frac{q_a}{q_{sat}(T_a)}\right) \left(\frac{U_5}{U_t}\right)^{b_{\mathrm{SUBL}}} \Delta t\right) \text{ where } U_t = \frac{-\log\left(\frac{\mathrm{MOB}_1 + 1}{c_{\mathrm{SUBL}}}\right)}{d_{\mathrm{SUBL}}}$$

$$\Delta m_1 = \min\left(0.5 m_1, a_{\mathrm{SUBL}} \left(\frac{T_0}{T_a}\right)^{\gamma_{\mathrm{SUBL}}} U_t \rho_a q_{sat}(T_a, P_s) \left(1 - \frac{q_a}{q_{sat}(T_a, P_s)}\right) \left(\frac{U_5}{U_t}\right)^{b_{\mathrm{SUBL}}} \Delta t\right) \tag{69}$$

$$\text{where } U_t = \frac{-\log\left(\frac{\mathrm{MOB}_1 + 1}{c_{\mathrm{SUBL}}}\right)}{d_{\mathrm{SUBL}}} \tag{70}$$

The air ~~volumetric mass~~ density ρ_a ($\mathrm{kg\,m^{-3}}$) and the saturation specific humidity $\cancel{q_{sat}(T_a)}$ $q_{sat}(T_a, P_s)$ ($\mathrm{kg\,kg^{-1}}$) are computed following Appendix F. a_{SUBL}, b_{SUBL}, γ_{SUBL}, c_{SUBL}, d_{SUBL} are dimensionless parameters defined in Appendix **??**

2.4.9 Absorption of solar radiation

Two schemes of different complexities are currently implemented and can be activated through the SNOWRAD option. When SNOWRAD=B92, the initial 3-band scheme of **?** is applied, inspired from **?**. The incoming solar radiation at the interface between layers i and $i+1$ is defined by:

$$R_i = \sum_{k=1}^{3} (1-\alpha_k)\gamma_k \mathrm{SW}\!\downarrow \exp\left(-\sum_{j=1}^{i} \beta_{k_j} z_j\right) \forall i \in [0,N] \quad \text{if SNOWRAD=B92} \tag{71}$$

The spectral partitioning of incoming radiation is fixed by parameters γ_k (Appendix **??**, with $\sum_{k=1}^{3} \gamma_k = 1$).

Spectral albedo values α_k are parameterized by:

$$\alpha_k = \chi \alpha_{k_1} + (1-\chi)\alpha_{k_2} \forall k \in [1,3] \tag{72}$$

where $$\chi = 0.8 \min\left(1, \frac{z_1}{0.02}\right) + 0.2 \min\left[1, \max\left(0, \frac{z_1 - 0.02}{0.01}\right)\right] \tag{73}$$

$$\begin{cases} \alpha_{1_i} & = \max\left[0.6, \min\left(0.92, 0.96 - 1.58\sqrt{d_i}\right) - 0.2\frac{A_i}{\tau_a} \times \min\left(\max\left(0.5, \frac{P_s}{P_{\text{CDP}}}\right), 1.5\right)\right] \text{ if } \rho_i < \rho_G \\ \alpha_{1_i} & = \alpha_{1_G} \text{ if } \rho_i \geq \rho_G \end{cases} \quad \text{for } i \in [1,2] \tag{74}$$

$$\begin{cases} \alpha_{2_i} & = \max\left(0.3, 0.9 - 15.4\sqrt{d_i}\right) \text{ if } \rho_i < \rho_G \\ \alpha_{2_i} & = \alpha_{2_G} \text{ if } \rho_i \geq \rho_G \end{cases} \quad \text{for } i \in [1,2] \tag{75}$$

$$\begin{cases} \alpha_{3_i} & = 346.3 d'_i - 32.31\sqrt{d'_i} + 0.88 \text{ if } \rho_i < \rho_G \quad \text{where} \quad d'_i = \min(d_i, 0.0023) \\ \alpha_{3_i} & = \alpha_{3_G} \text{ if } \rho_i \geq \rho_G \end{cases} \quad \text{for } i \in [1,2] \tag{76}$$

Note that compared to **?**, the consideration of optical diameters of the first two layers, already in the code but not documented, is now made explicit in Eq. 72. The time constant τ_a in Eq. 74 is the main control of the parameterization reducing snow albedo in the visible band as a function of the age of the layer A_i in order to mimic the effect of Light-Absorbing Particles (LAP). Its default value is set to 60 days with an elevation-dependent multiplicative correction factor (function of P_s) assuming that LAP deposition decreases with elevation. However, it is recommended to adjust this parameter depending on the expected amount of LAP in the target region (**?**), to consider calibration against observed albedo time series when possible, or to apply several values of this parameter in multiphysics applications (**?**). ~~Last modification ,~~ The last modification about absorption of solar radiation consists in applying constant glacier spectral albedo values α_{k_G} on snow-free glacier surfaces. Default values (Appendix **??**) are taken from **?** but must be adjusted to each specific glacier (e.g. **?**)

Absorption coefficients β_{k_i} (m^{-1}) for band k and layer i are parameterized by:

$$\beta_{1_i} = \max\left(40, 0.00192\frac{\rho_i}{\sqrt{d_i}}\right) \forall i \in [1,N] \tag{77}$$

$$\beta_{2_i} = \max\left(100, 0.01098\frac{\rho_i}{\sqrt{d_i}}\right) \forall i \in [1,N] \tag{78}$$

$$\beta_{3_1} = +\infty \tag{79}$$

Alternatively to Eq. 71, an option is available for solar radiative transfer calculation in the snowpack (SNOWRAD=T17) combining the TARTES radiative scheme (Two-streAm Radiative TransfEr in Snow model, **?**) and an explicit modelling of LAP (**?**). TARTES is a two-stream radiative transfer scheme based on an analytical formulation of radiative transfer in snow (**?**). The scheme is applied separately for the direct (DIR) and diffuse (DIF) components of solar radiation. For both components, TARTES computes spectral solar absorption within each layer E_{k_i} (Eq. 59-63 in the comprehensive scientific documentation of TARTES, **?**) and the spectral albedo α_k (Eq. 63 in **?**). Eq. 80 is used to come back to the profile of R_i used in Crocus:

$$R_i = \sum_{k=1}^{N_k}\left((1-\alpha_{k\ \mathrm{DIR}})\gamma_{k\ \mathrm{DIR}}\mathrm{SW}_{\mathrm{DIR}}{\downarrow} - \sum_{j=1}^{i} E_{k_j\mathrm{DIR}} + (1-\alpha_{k\ \mathrm{DIF}})\gamma_{k\ \mathrm{DIF}}\mathrm{SW}_{\mathrm{DIF}}{\downarrow} - \sum_{j=1}^{i} E_{k_j\mathrm{DIF}}\right) \forall i \in [0,N] \quad \text{if SNOWRAD=T17} \tag{80}$$

The default spectral resolution is 20 nm for $N_k = 111$ spectral bands in the interval [300 nm - 2500 nm]. Coefficients $\gamma_{k\ \mathrm{DIR}}$ and $\gamma_{k\ \mathrm{DIF}}$ to split input direct and diffuse broadband radiation in spectral solar irradiance are currently provided as fixed parameters derived from SBDART (**?**) under the conditions encountered at Col de Porte~~site~~.

To compute E_{k_i}, TARTES accounts for the effect of snow physical properties (SSA and density) and Light-Absorbing Particles using the mass absorption efficiency of each LAP type and its mass content vertical profile. For dust, the mass absorption efficiency is defined following Eq. 83 of **?** with parameters $\lambda_0 = 400\ \mathrm{nm}$, MAE$(\lambda_0) = 110\ \mathrm{m}^2\mathrm{kg}^{-1}$ and AAE $= 4.1$ (values for dust PM2.5 from Libya in Table 4 of **?**). For black carbon, the mass absorption efficiency is defined following Eq. 82 of **?** with a constant density $\rho_{\mathrm{BC}} = 1270\ \mathrm{kg\,m}^{-3}$ (**?**) and a constant refractive index ($m_{\mathrm{BC}} = 1.95 - 0.79i$ from **?**). The MAE is then scaled with a multiplicative factor $f_{\mathrm{BC}} = 1.638$ to obtain an MAE value at 550 nm of $1.125 \times 10^4\ \mathrm{m}^2\mathrm{kg}^{-1}$ consistently with measurements of **?**. The scaling makes it possible to implicitly account for the potential absorption enhancement due to internal particle mixing or particle coating. The implementation of new types of LAP would require the implementation of the description of their associated mass absorption efficiency. The vertical profile of LAP mass content is obtained following Section 2.4.10.

2.4.10 Light-Absorbing Particles

From **?**, the evolution law of the mass $\mathcal{M}_{i,j}$ of LAP j in layer i is expressed as follows:

$$\mathcal{M}^{+}_{1,j} = \mathcal{M}^{-}_{1,j} + \mathcal{W}_{w,j}\Delta t + \frac{\mathcal{W}_{d,j}\Delta t \exp\left(\frac{-z_1}{h}\right)}{\sum_{k=1}^{N}\exp\left(\frac{-Z_k}{h}\right)} \tag{81}$$

$$\mathcal{M}^{+}_{i,j} = \mathcal{M}^{-}_{i,j} + \frac{\mathcal{W}_{d,j}\Delta t \exp\left(\frac{-Z_i}{h}\right)}{\sum_{k=1}^{N}\exp\left(\frac{-Z_k}{h}\right)} \forall i \in [2,N] \tag{82}$$

where Z_i represents the depth of layer i from the surface ($Z_i = \sum_{j=1}^{i} z_j$) and h the e-folding depth characterizing the decrease rate with depth of the impact of the dry deposition flux $\mathcal{W}_{d,j}$. The wet deposition flux $\mathcal{W}_{w,j}$ only affects the uppermost layer.

However, in case of thin layers at the surface, ~~an~~ a homogeneous repartition is ~~then~~ applied on the uppermost N_{10} layers gathering the first 10 $\mathrm{kg\,m^{-2}}$ of snow. This limits artificial albedo variations due to vertical regridding (**?**):

$$\mathcal{M}^{+}_{i,j} = \frac{m_i}{\sum_{k=1}^{N_{10}} m_k} \times \sum_{k=1}^{N_{10}} \mathcal{M}^{+}_{k,j} \forall i \in [1, N_{10}] \text{ where } N_{10} \cancel{\text{is the highest integer such as}} \text{ is the highest integer such that } \sum_{k=1}^{N_{10}-1} m_k < 10 \text{ kg m}^{-} \tag{83}$$

Note than the wet deposition flux should be consistent with the occurrence of precipitation in the model input data. If not, any positive wet deposition flux without precipitation is not incorporated ~~to~~ in the snowpack LAP mass content (**?**).

2.4.11 Turbulent fluxes

The sensible and latent turbulent heat fluxes are expressed by:

$$H = \rho_a c_P C_H U_{>1}\left(\frac{T_1}{\Pi_s} - \frac{T_a}{\Pi_a}\right) \tag{84}$$

$$LE = \rho_a L_s C_H U_{>1}\left(q_{sat}(T_1, P_s) - q_a\right) \tag{85}$$

ρ_a is the air volumetric mass ($\mathrm{kg\,m^{-3}}$), Π_s and Π_a the Exner functions at the surface and at the level of the atmospheric forcing and ~~$q_{sat}(T_1)$~~ $q_{sat}(T_1, P_s)$ the saturation specific humidity at surface temperature T_1 ($\mathrm{kg\,kg^{-1}}$), cf. Appendix F for their computations. A minimum value of 1 $\mathrm{m\,s^{-1}}$ is applied to wind speed ($U_{>1} = \max(U, 1\ \mathrm{m\,s^{-1}})$) to maintain minimal fluxes even with very low wind speeds. Note that a simplification has been introduced compared to **?** by considering only surface sublimation and not evaporation of liquid water. The exchange coefficient C_H depends on the stability of the atmosphere through the Richardson Number Ri (Appendix F) following **?**:

$$\begin{cases} C_H = \frac{\kappa_{\mathrm{VK}}^2}{\ln\left(\frac{z_u}{z_0}\right)\ln\left(\frac{z_a}{z_{0_h}}\right)}\left(1 - \frac{15\mathrm{Ri}}{1 + C_h\sqrt{|\mathrm{Ri}|}}\right) & \text{if Ri} <= 0 \\ C_H = \frac{\kappa_{\mathrm{VK}}^2}{\ln\left(\frac{z_u}{z_0}\right)\ln\left(\frac{z_a}{z_{0_h}}\right)} \times \frac{1}{1 + 15\mathrm{Ri}^*\sqrt{1 + 5\mathrm{Ri}^*}} & \text{if Ri} > 0 \text{ where } \mathrm{Ri}^* = \min(\mathrm{Ri}, \mathrm{Ri}_l) \end{cases} \tag{86}$$

where κ_{VK} is the Von Karman constant. Coefficient C_h in the unstable case is defined by:

$$C_h = 15\left(3.2165 + 4.3431\ln\left(\frac{z_0}{z_{0_h}}\right) + 0.5360\left(\ln\left(\frac{z_0}{z_{0_h}}\right)\right)^2 - 0.0781\left(\ln\left(\frac{z_0}{z_{0_h}}\right)\right)^3\right)\frac{k^2}{\ln\left(\frac{z_u}{z_0}\right)\ln\left(\frac{z_a}{z_{0_h}}\right)}\left(\frac{z_a}{z_{0_h}}\right)^p \tag{87}$$

$$\text{where } p = 0.5802 - 0.1571\ln\left(\frac{z_0}{z_{0_h}}\right) + 0.0327\left(\ln\left(\frac{z_0}{z_{0_h}}\right)\right)^2 - 0.0026\left(\ln\left(\frac{z_0}{z_{0_h}}\right)\right)^3 \tag{88}$$

In the stable case, an adjustable threshold Ri_l is applied on the Richardson number (**?**). Considering the high uncertainty of these parameterizations of turbulent processes, sensitivity analyses to momentum and thermodynamic roughness lengths z_0 and z_{0_h} (m) and to parameter Ri_l are recommended for robust applications of the model (**?**).

2.4.12 Heat diffusion and energy balance

The formalism of this section is largely inspired by the documentation of the ISBA-ES (Explicit Snow) snowpack scheme from **?**. The model solves the heat diffusion in the stratified snowpack using an implicit time integration scheme:

$$c_I\rho_i z_i\frac{T_i^+ - T_i^-}{\Delta t} + E_i = G_{i-1}^+ - G_i^+\ \forall i\in[1,N] \tag{89}$$

where G_i^+ ($\mathrm{Wm^{-2}}$) represents the heat flux between layers i and $i+1\ \forall i\in[1,N-1]$ at the end of the time step. G_0^+ and G_N^+ are the heat fluxes at the interfaces with atmosphere and soil at the end of the time step and E_i ($\mathrm{Wm^{-2}}$) the energy of phase change for layer i. The main assumption of the model is that it is possible to separate heat diffusion and phase changes. Thus, the diffusion is solved assuming $E_i = 0\ \forall i\in[1,N]$.

The heat flux between two snow layers i and $i+1$ is the sum between the radiative flux R_i and the heat conduction K_i^+ ($\mathrm{Wm^{-2}}$) between these layers:

$$G_i^+ = K_i^+ + R_i\ \forall i\in[1,N-1] \tag{90}$$

The conduction flux K_i is expressed at the end of the time step by:

$$K_i^+ = 2\overline{\lambda_i}\frac{T_i^+ - T_{i+1}^+}{z_i + z_{i+1}}\ \forall i\in[1,N-1] \tag{91}$$

where $\overline{\lambda_i}$ is the harmonic weighted mean of the thermal conductivities of layers i and $i+1$:

$$\overline{\lambda_i} = \cancel{\frac{z_i\lambda_i + z_{i+1}\lambda_{i+1}}{z_i + z_{i+1}}}\ \frac{z_i + z_{i+1}}{\frac{z_i}{\lambda_i} + \frac{z_{i+1}}{\lambda_{i+1}}} \tag{92}$$

~~Note that some models (including ISBA-ES) rather use a harmonic weighted mean although~~ Eq. 92 replaces the arithmetic mean used in previous versions of Crocus (**?**). Although there is not a clear agreement in the literature that ~~it outperforms~~ harmonic means outperform the modelling of heat diffusion (**?**), this solution is more consistent with the other components of SURFEX and more commonly used today in snow models (**?**).

The thermal conductivity λ_i of layer i is parameterized as a function of density ρ_i following parameterizations of **?**, **?** or **?**:

$$\begin{cases} \text{If SNOWCOND=Y81:} & \lambda_i = \max\left[a_\lambda(\frac{\rho_i}{\rho_w})^{1.88}\,;\,\lambda_{\min}\right] \\ \text{If SNOWCOND=C11:} & \lambda_i = b_\lambda \rho_i^2 + c_\lambda \rho_i + d_\lambda \\ \text{If SNOWCOND=I02:} & \lambda_i = e_\lambda + f_\lambda \rho_i^2 + \left(g_\lambda + \frac{h_\lambda}{T_i^- + i_\lambda}\right)\frac{P_0}{P} \end{cases} \tag{93}$$

All empirical parameters are provided in Appendix **??**. Alternatives to Eq. 93 were proposed by **?** for Arctic snow. Although not available in version 3.0.1, they will be implemented in a future version.

The heat flux between the atmosphere and the surface is the sum of all energy fluxes at the surface:

$$G_0 = R_0 + \epsilon(\text{LW}{\downarrow} - \sigma T_1^4) - H - LE + \mathcal{P}_r \Delta t \times c_W (T_a - T_0) + E_{\text{FRZ}} \tag{94}$$

All fluxes in ~~Equation~~ Eq. 94 are expressed at time $t + \Delta t$ with the following approximation:

$$F^+ = F^- + \frac{\partial F}{\partial T_1}(T_1^+ - T_1^-) \tag{95}$$

Thus,

$$\begin{aligned} G_0^+ =& R_0 + \epsilon(\text{LW}{\downarrow} - \sigma (T_1^-)^3(4T_1^+ - 3T_1^-)) - \rho_a c_P C_H U_{>1}(\frac{T_1^+}{\Pi_s} - \frac{T_a}{\Pi_a}) \\ & - \rho_a L_s C_H U_{>1}\left(q_{sat}(T_1^-, P_s) - q_a + \frac{\partial q_{sat}}{\partial T_1}(T_1^+ - T_1^-)\right) + \mathcal{P}_r \Delta t \times c_W (T_a - T_0) + E_{\text{FRZ}} \end{aligned} \tag{96}$$

The computation of $\frac{\partial q_{sat}}{\partial T_1}$ is detailed in Appendix F. The heat flux between the bottom layer and the ground G_N^+ is expressed with a semi-implicit coupling (i.e. considering the ground surface temperature $T_{G_1}^-$ at time step t) in order to solve separately the thermal diffusion in the soil, outside the snowpack model:

$$G_N^+ = 2\overline{\lambda_N}\frac{T_N^+ - T_{G_1}^-}{z_N + z_{G_1}} + R_N \tag{97}$$

where $\overline{\lambda_N}$ is the harmonic mean between the thermal conductivity of the bottom snow layer λ_N and the thermal conductivity of the first soil layer λ_{G_1}:

$$\overline{\lambda_N} = \frac{z_N + z_{G_1}}{\frac{z_N}{\lambda_N} + \frac{z_{G_1}}{\lambda_{G_1}}} \tag{98}$$

Combining and rearranging Eq. 89-91, 96 and 97 the system to solve becomes:

$$\begin{cases} \begin{cases} \left(\frac{c_I\rho_1 z_1}{\Delta t} + \frac{2\overline{\lambda_1}}{z_1+z_2} + 4\epsilon\sigma(T_1^-)^3 + \rho_a C_H U_{>1}\left(\frac{c_P}{\Pi_s} + L_s\frac{\partial q_{sat}}{\partial T_1}\right)\right)T_1^+ - \frac{2\overline{\lambda_1}}{z_1+z_2}T_2^+ = \\ \frac{c_I\rho_1 z_1}{\Delta t}T_1^- + R_0 - R_1 + \epsilon\left(\text{LW}{\downarrow} + 3\sigma(T_1^-)^4\right) + \rho_a C_H U_{>1}\left(c_P\frac{T_a}{\Pi_a} + L_s\left(q_a - q_{sat}(T_1^-, P_s) + \frac{\partial q_{sat}}{\partial T_1}T_1^-\right)\right) \\ +\mathcal{P}_r \Delta t \times c_W (T_a - T_0) + E_{\text{FRZ}} \end{cases} \\ -\frac{2\overline{\lambda_{i-1}}}{z_{i-1}+z_i}T_{i-1}^+ + \left(\frac{c_I\rho_i z_i}{\Delta t} + \frac{2\overline{\lambda_i}}{z_i+z_{i+1}} + \frac{2\overline{\lambda_{i-1}}}{z_{i-1}+z_i}\right)T_i^+ - \frac{2\overline{\lambda_i}}{z_i+z_{i+1}}T_{i+1}^+ = \frac{c_I\rho_i z_i}{\Delta t}T_i^- + R_{i-1} - R_i\; \forall i \in [2, N-1] \\ -\frac{2\overline{\lambda_{N-1}}}{z_{N-1}+z_N}T_{N-1}^+ + \left(\frac{c_I\rho_N z_N}{\Delta t} + \frac{2\overline{\lambda_N}}{z_N+z_{G_1}} + \frac{2\overline{\lambda_{N-1}}}{z_{N-1}+z_N}\right)T_N^+ = \frac{2\overline{\lambda_N}}{z_N+z_{G_1}}T_{G_1}^- + \frac{c_I\rho_N z_N}{\Delta t}T_N^- + R_{N-1} - R_N \end{cases} \tag{99}$$

The first line of Eq. 99 is rewritten in the form:

$$\frac{1}{C_T \Delta t} T_1^+ - \frac{\zeta_2}{C_T \Delta t} T_2^+ = \frac{\zeta_1}{C_T \Delta t} \tag{100}$$

$$\text{where } C_T = \frac{1}{c_I \rho_1 z_1} \tag{101}$$

$$\zeta_2 = \frac{2 C_T \overline{\lambda_1}}{\mathcal{A}(z_1 + z_2)} \tag{102}$$

$$\mathcal{A} = \frac{1}{\Delta t} + C_T \left[\frac{2\overline{\lambda_1}}{z_1 + z_2} + 4\epsilon\sigma (T_1^-)^3 + \rho_a C_H U_{>1} \left(\frac{c_P}{\Pi_s} + L_s \frac{\partial q_{sat}}{\partial T_1} \right) \right] \tag{103}$$

$$\zeta_1 = \frac{\mathcal{B} T_1^- + \mathcal{C}}{\mathcal{A}} \tag{104}$$

$$\mathcal{B} = \frac{1}{\Delta t} + C_T \left[3\epsilon\sigma (T_1^-)^3 + \rho_a C_H U_{>1} \frac{\partial q_{sat}}{\partial T_1} \right] \tag{105}$$

$$\mathcal{C} = C_T \left[R_0 - R_1 + \epsilon \text{LW}\!\downarrow + \rho_a C_H U_{>1} \left(c_P \frac{T_a}{\Pi_a} + L_s \left(q_a - q_{sat}(T_1^-, P_s) \right) \right) + \mathcal{P}_r \Delta t \times c_W (T_a - T_0) + E_{\text{FRZ}} \right] \tag{106}$$

Therefore, the temperature profile at time $t + \Delta t$ is obtained by:

$$\begin{bmatrix} T_1^+ \\ T_2^+ \\ ... \\ T_i^+ \\ ... \\ ... \\ T_N^+ \end{bmatrix} = \begin{bmatrix} B_1 & C_1 & 0 & 0 & ... & ... & 0 \\ A_2 & B_2 & C_2 & 0 & ... & ... & 0 \\ 0 & A_3 & ... & ... & ... & ... & ... \\ 0 & 0 & ... & B_i & C_i & ... & 0 \\ ... & ... & ... & A_i & ... & ... & 0 \\ ... & ... & ... & ... & ... & ... & C_{N-1} \\ 0 & 0 & ... & 0 & 0 & A_N & B_N \end{bmatrix}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \\ ... \\ Y_i \\ ... \\ ... \\ Y_N \end{bmatrix} \tag{107}$$

where vectors $A=(A_2,...,A_i,...A_N)$, $B=(B_1,...,B_i,...B_N)$, $C=(C_1,...,C_i,...C_{N-1})$, $Y=(Y_1,...,Y_i,...Y_N)$ are defined by:

$$\begin{cases} A_i = & -\frac{2\overline{\lambda_{i-1}}}{z_{i-1}+z_i}\forall i\in[2,N] \\ B_1 = & \frac{1}{C_T\Delta t} \\ B_i = & \left(\frac{c_I\rho_i z_i}{\Delta t}+\frac{2\overline{\lambda_i}}{z_i+z_{i+1}}+\frac{2\overline{\lambda_{i-1}}}{z_{i-1}+z_i}\right)\forall i\in[2,N-1] \\ B_N = & \left(\frac{c_I\rho_N z_N}{\Delta t}+\frac{2\overline{\lambda_N}}{z_N+z_{G_1}}+\frac{2\overline{\lambda_{N-1}}}{z_{N-1}+z_N}\right) \\ C_1 = & -\frac{\zeta_2}{C_T\Delta t} \\ C_i = & -\frac{2\overline{\lambda_i}}{z_i+z_{i+1}}\forall i\in[2,N-1] \\ Y_1 = & \frac{\zeta_1}{C_T\Delta t} \\ Y_i = & \frac{c_I\rho_i z_i}{\Delta t}T_i^- + R_{i-1}-R_i\forall i\in[2,N-1] \\ Y_N = & \frac{2\overline{\lambda_N}}{z_N+z_{G_1}}T_{G_1}^- + \frac{c_I\rho_N z_N}{\Delta t}T_N^- + R_{N-1}-R_N \end{cases} \tag{108}$$

2.4.13 Adjustments in case of surface melting

The possibility for T_1^+ to exceed the freezing point in the solving of Eq. 107 can lead to overestimate the surface energy fluxes that depend on T_1^+ and to overestimate the heat conduction K_1 below surface. To avoid this numerical artefact, Crocus distinguishes the case of first melting ($T_1^- < T_0$ and $T_1^+ > T_0$), and ongoing melting ($T_1^- \geq T_0$ and $T_1^+ > T_0$).

In the case of a first melting ($T_1^- < T_0$ and $T_1^+ > T_0$), the temperature profile for layers $i \in [2,N]$ is not updated, so only the surface fluxes are adjusted replacing T_1^+ by T_0 in Eq. 84, 85, 96. The new temporary surface temperature (before melting) is obtained by converting the difference in both consecutive estimates of the surface energy flux G_0 in terms of temperature change:

$$\text{If } T_1^- < T_0 \text{ and } T_1^+ > T_0: \begin{cases} G_0^+ = & R_0+\epsilon(\mathrm{LW}{\downarrow} - \sigma(T_1^-)^3(4T_0-3T_1^-)) - \rho_a c_P C_H U_{>1}(T_0-T_a) \\ & -\rho_a L_s C_H U_{>1}\left(q_{sat}(T_1^-,P_s)-q_a+\frac{\partial q_{sat}}{\partial T_0}(T_0-T_1^-)\right)+\mathcal{P}_r\Delta t\times c_W(T_a-T_0) \\ T_1^+ = & T_1^{\text{first}}+\frac{\Delta t}{c_I\rho_i z_i}(G_0^+-G_0^{\text{first}}) \end{cases} \tag{109}$$

where T_1^{first} is the solution of Eq. 107, and G_0^{first} is the flux obtained applying Eq. 96 with T_1^{first}. This ensures that the energy budget over the snowpack is closed.

In the case of an ongoing melting (i.e. the solution of Eq. 107 provides a temperature above freezing point for the surface layer at two consecutive time steps: $T_1^- \geq T_0$ and $T_1^+ > T_0$), the system is solved a second time for layers $i \in [2,N]$ by constraining $T_1^+ = T_0$:

$$\begin{bmatrix} T_2^+ \\ ... \\ T_i^+ \\ ... \\ ... \\ T_N^+ \end{bmatrix} = \begin{bmatrix} B_2 & C_2 & 0 & ... & ... & 0 \\ A_3 & ... & ... & ... & ... & ... \\ 0 & ... & B_i & C_i & ... & 0 \\ ... & ... & A_i & ... & ... & 0 \\ ... & ... & ... & ... & ... & C_{N-1} \\ 0 & ... & 0 & 0 & A_N & B_N \end{bmatrix}^{-1} \begin{bmatrix} Y_2 + \frac{2\overline{\lambda_1}}{z_1+z_2} T_0 \\ ... \\ Y_i \\ ... \\ ... \\ Y_N \end{bmatrix} \tag{110}$$

Then, the surface energy fluxes are updated replacing T_1^+ and T_1^- by T_0 in Eq. 84, 85, 96 and the conduction flux between the two first layers is updated with the solution of Eq. 110. A new estimate of the surface layer T_1^+ is temporarily obtained from Eq. 89 ($i = 1$) consistently with these updated fluxes, before the transfer of exceeding energy in phase change (E_1, see Section 2.4.16):

$$\text{If } T_1^- \geq T_0 \text{ and } T_1^+ > T_0 : \begin{cases} G_0^+ = & R_0 + \epsilon(\text{LW}\!\downarrow - \sigma T_0^4) - \rho_a c_P C_H U_{>1}(T_0 - T_a) \\ & -\rho_a L_s C_H U_{>1}\left(q_{sat}(T_0, P_s) - q_a\right) + \mathcal{P}_r \Delta t \times c_W (T_a - T_0) \\ K_1^+ = & 2\overline{\lambda}_i \frac{T_0 - T_2^+}{z_i + z_{i+1}} \\ T_1^+ = & T_0 + \frac{\Delta t}{c_I \rho_i z_i}\left(G_0^+ - K_1^+ - R_1\right) \end{cases} \tag{111}$$

This also guarantees the closure of the energy budget in that case. Note that **?** recently proposed alternative model formulations to compute a more stable surface energy balance with a better coupling between surface melting and heat transfers. This should be explored in the future to avoid the need of such numerical adjustments.

2.4.14 Coupling with MEB

MEB (Multiple Energy Balance, **?**) is a variant of the ISBA land surface scheme in which the soil-vegetation system is no longer described by a composite approach but by an explicit representation of vegetation with a big-leaf approach. This allows ~~to represent~~ representation of the main physical processes involved in forest-snow interactions including snowfall interception and radiative impacts of the trees. An extensive description of this implementation is beyond the scope of this paper but available in **?**. However, as coupled processes require a coupled solving of snow surface and vegetation temperatures T_1 and T_v, the solving of heat diffusion is modified when Crocus is coupled to the MEB scheme. In that case, the values of the surface energy fluxes are no longer obtained through the implicit solving of Eq. 99-107 but are imposed as a boundary condition to conserve energy. Eq. 89 is modified for the surface layer to maintain the fluxes obtained from MEB:

$$c_I \rho_i z_i \frac{T_1^+ - T_1^-}{\Delta t} + E_1 = G_{0_{\text{MEB}}} - R_1 - K_1^+ \tag{112}$$

$$\text{where } G_{0_{\text{MEB}}} = R_0 + \epsilon(\text{LW}\!\downarrow - \sigma T_{1_{\text{MEB}}}^4) - H_{\text{MEB}} - LE_{\text{MEB}} + \mathcal{P}_r \Delta t \times c_W (T_a - T_0) \tag{113}$$

$G_{0_{\text{MEB}}}$ represents the total surface energy flux in agreement with the first estimate of surface temperature from MEB $T_{1_{\text{MEB}}}$ and the associated turbulent fluxes H_{MEB} and LE_{MEB}. LW↓ accounts for vegetation radiation and is therefore linked to the solution obtained for vegetation temperature $T_{v_{\text{MEB}}}$. It must be noted that as R_0 and R_1 are already needed in MEB, they are computed with the same routine as described in Section 2.4.9 but earlier in the time step (before the MEB solving) to guarantee identical values at both steps (MEB and Crocus), and accounting for shading by trees in SW↓.

Consistently with Eq. 112, the first line of Eq. 99 is replaced by:

$$\left(\frac{c_I \rho_1 z_1}{\Delta t} + \frac{2\overline{\lambda_1}}{z_1+z_2}\right) T_1^+ - \frac{2\overline{\lambda_1}}{z_1+z_2} T_2^+ = \frac{c_I \rho_1 z_1}{\Delta t} T_1^- + G_{0_{\text{MEB}}} - R_1 \tag{114}$$

Rewriting Eq. 114 with the same form as Eq. 100 is equivalent to replacing Eq.103, 105, 106 by ~~:~~

$$\mathcal{A} = \frac{1}{\Delta t} + C_T\left[\frac{2\overline{\lambda_1}}{z_1+z_2}\right]$$

$$\mathcal{B} = \frac{1}{\Delta t}$$

$$\mathcal{C} = C_T\left[G_{0_{\text{MEB}}} - R_1\right]$$

~~and to~~ Eq. 115, 116, 117 and solving the linear system of Eq. 107 with coefficients C_1 and Y_1 of Eq. 108 modified ~~accordingly~~ according to the new values of ζ_2 and ζ_1 from Eq. 102 and Eq. 104.

$$\mathcal{A} = \frac{1}{\Delta t} + C_T\left[\frac{2\overline{\lambda_1}}{z_1+z_2}\right] \tag{115}$$

$$\mathcal{B} = \frac{1}{\Delta t} \tag{116}$$

$$\mathcal{C} = C_T\left[G_{0_{\text{MEB}}} - R_1\right] \tag{117}$$

The adjustments of Section 2.4.13 are no longer required as G_0 is imposed by MEB and phase change was already accounted for to compute this flux.

~~Sequence of Crocus subroutines for each time step without (blue) or with (green) coupling with MEB. Optional subroutines in light blue.~~

The validity of Eq. 114 would require that no modification of the state variables of the snowpack has occurred between the computation of $G_{0_{\text{MEB}}}$ and the solving of Eq. 107 because $G_{0_{\text{MEB}}}$ depends on snow properties through Eq. 48 and 49 in **?** and the associated Appendix I4. In practice, the SURFEX code structure makes this constraint especially challenging. The violation of this assumption can generate numerical instabilities especially with thin surface layers due to the violation of the second principle of thermodynamics (?). To reduce as much as possible the occurrence of this problem, the sequence of routines is modified following Figure 1. However, there is currently no solution to avoid modifications due to snow interception by vegetation because this term ~~can not~~ cannot be computed before ~~MEB solving~~ solving MEB (the mass balance depends on latent heat terms known only after the solving). Numerical instabilities are therefore still possible and further investigations are in progress to safely allow large scale applications of MEB-Crocus. This issue is more challenging than in the case of the coupling of MEB with ES snow scheme (**??**) which has already been successfully applied in large scale simulations. This is probably due to the possible occurrence of thinner surface snow layers with Crocus.

2.4.15 Total melting or sublimation

The snowpack is assumed to totally disappear when

$$G_0^+ - G_N^+ \geq \frac{-\sum_{i=1}^{N^t} H_i^+ + H_{\mathrm{subl}}}{\Delta t} \text{ or } \frac{\max(0, LE^+)\Delta t}{L_s} \geq \sum_{i=1}^{N^t} m_i \tag{118}$$

The first condition corresponds to a total melt of the snowpack (energy gain during the time step exceeds available internal energy for melting). It requires to remove the enthalpy of surface snow potentially sublimated H_{subl} from the total enthalpy. The second condition corresponds to a total sublimation of the snowpack, which is a very unusual case of snow disappearance but can appear especially when a snow transport module provide a very low amount of snow on bare ground in cold and windy conditions.

In both cases, for mass conservation, the runoff at the bottom of the snowpack is set to the total snow mass and the energy associated with LE^+ is transmitted to the soil scheme as a correction term. If Eq. 118 is not verified, the procedure follows Sect. 2.4.16 to Sect. 2.4.20.

2.4.16 Melting

Let's define the mass of solid and liquid fractions of the snow before any phase change s_i^- and l_i^- (kg m^{-2}):

$$s_i^- = m_i - l_i^- \tag{119}$$

When the heat diffusion solving provides a temperature T_i^+ above the melting point T_0 for layer i, the model simulates melting. The energy available for fusion E_{f_i} (J m^{-2}) on layer i can be constrained either by the ~~heating energy either~~ available heating energy or by the available mass of the solid fraction of snow before melting s_i^-. Thus, the energy and mass f_i (kg m^{-2}) of melting during the time step are computed by:

$$E_{f_i} = \min\left(c_I \rho_i (T_i^+ - T_0), L_m s_i^-\right) \tag{120}$$

$$f_i = \frac{E_{f_i}}{L_m} \tag{121}$$

The mass of solid and liquid fractions after melting are:

$$s_i^+ = s_i^- - f_i \tag{122}$$

$$l_i^+ = l_i^- + f_i \tag{123}$$

The corresponding updates of depth and total density can be expressed by:

$$z_i^+ = z_i^- \times \frac{m_i - l_i^+}{m_i - l_i^-} \tag{124}$$

$$\rho_i^+ = \frac{m_i}{z_i^+} \tag{125}$$

In case of melting, the melting point temperature is attributed to the layer temperature:

$$T_i^{++} = \min(T_0, T_i^+) \tag{126}$$

In practice, a first evaluation of Eq.120-121 is computed for all layers to identify cases where a numerical layer fully melts out ($c_I \rho_i (T_i^+ - T_0) \geq L_m s_i^-$). In such a case, the numerical layer i is aggregated with the numerical layer $i+1$ for $i \in [1, N-1]$, following Equations 12-20. If the bottom layer N is concerned, it is agregated with the above layer $N-1$. Several iterations can be done in case of melting of multiple consecutive layers. Then, the melting is computed again with this updated discretization with the guarantee that all numerical layers remain defined.

2.4.17 Refreezing

When the heat diffusion solving provides a temperature T_i^+ below the melting point T_0 whereas liquid water is present, the model simulates refreezing. The energy available for refreezing E_{r_i} (J m^{-2}) can be constrained either by the layer cooling after diffusion ~~either~~ or by the maximum available liquid water content before refreezing l_i^-. Thus, the energy and mass r_i (kg m^{-2}) of refreezing during the time step are computed by:

$$E_{r_i} = \min(c_I \rho_i (T_0 - T_i^+), L_m l_i^-) \tag{127}$$

$$r_i = \frac{E_{r_i}}{L_m} \tag{128}$$

The mass of solid and liquid fractions after refreezing are:

$$s_i^+ = s_i^- + r_i \tag{129}$$

$$l_i^+ = l_i^- - r_i \tag{130}$$

The energy conservation during the refreezing process is expressed by:

$$c_I \rho_i (T_i^+ - T_i^{++}) s_i^+ + E_{r_i} + c_I \rho_i (T_0 - T_i^+) r_i = 0 \tag{131}$$

where T_i^+ and T_i^{++} (K) are the layer temperature of the solid fraction before and after refreezing. In Eq. 131, the first term corresponds to the heating of the solid fraction after refreezing, the second term to the latent heat release due to refreezing and the third term to the cooling of the refrozen part necessary for the thermal equilibrium of the solid phase.

By combining Eq. 131 and 130, the evolution of the layer temperature due to refreezing is computed by:

$$T_i^{++} = T_0 + (T_i^+ - T_0) \frac{s_i^-}{s_i^+} + \frac{E_{r_i}}{c_I \rho_i s_i^+} \tag{132}$$

Equations 127-132 are not applied independently but jointly with liquid water percolation as described in Section 2.4.18 within Algorithm 1

2.4.18 Liquid water percolation

Let's define the volumetric liquid water content w_i ($\mathrm{kg\,m^{-3}}$):

$$w_i = \frac{l_i}{z_i} \tag{133}$$

and the snow porosity:

$$\phi_i = 1 - \frac{\rho_i^- - w_i}{\rho_I} \tag{134}$$

where ρ_I is the volumetric mass of pure ice. The liquid water flow $\mathcal{F}_i$ ($\mathrm{kg\,m^{-2}}$) between layers i and $i+1$ is computed by a simple and conceptual bucket approach where the layers are seen as superposed water reservoirs with a maximum liquid water holding capacity $w_{i\ \max}$. The excess water drains to the underlying layer when w_i exceeds $w_{i\ \max}$ following Algorithm 1. Snow layer densities ρ_i are updated with the resulting net mass flux $\mathcal{F}_i - \mathcal{F}_{i-1}$. However, in the limit case of ice layers ($\rho = \rho_I$) ~~, in case of refreezing~~ and refreezing ($r_i > 0$), the excess liquid water mass increases the ice layer depth at constant density. Note that another choice would be possible (**?**) by considering ice layers impermeable (i.e. computing $\mathcal{F}_i$ before any refreezing and r_i after percolation) with potential impacts on glaciers simulations.

Algorithm 1 Buckets algorithm for liquid water percolation

$\mathcal{F}_0 = \mathcal{P}_r \Delta t$
for $i \in [1, N]$ **do**
  Compute r_i, s_i, w_i with Eq. 127-133
  $w_i^* = w_i + \frac{\mathcal{F}_{i-1} + f_i - r_i}{z_i}$
  $\mathcal{F}_i = \max(0, w_i^* - w_{i\ \max}) z_i$
  $w_i = w_i^* - \frac{\mathcal{F}_i}{z_i}$
  $\rho_i^+ = \rho_i^- (\mathcal{F}_i - \mathcal{F}_{i-1}) \frac{\rho_W}{z_i}$
  if $\rho_i^+ > \rho_I$ **then**
    $z_i = \frac{\rho_i^+}{\rho_I} z_i$
    $\rho_i^+ = \rho_I$
  end if
end for

Several formulations of $w_{i\ \max}$ were implemented by **?**:

$$\begin{cases} \text{If SNOWLIQ = B92:} & w_{i\ \max} = 0.05 \rho_w \phi_i \\ \text{If SNOWLIQ = SPK:} & \begin{cases} w_{i\ \max} = \rho_w \left(0.08 - 0.1023(0.97 - \phi_i)\right) & \text{if } \phi_i \geq 0.77 \\ w_{i\ \max} = \rho_w \left(0.0264 + 0.0099 \frac{\phi_i}{1-\phi_i}\right) & \text{otherwise} \end{cases} \\ \text{If SNOWLIQ = B02:} & w_{i\ \max} = \rho_i \left(r_{\min} + (r_{\max} - r_{\min}) \max\left(0, \frac{\rho_r - \rho_i}{\rho_r}\right)\right) \end{cases} \tag{135}$$

Parameters $r_{\min}$, $r_{\max}$ ρ_r are defined in Appendix **??**.

The resolution of Richards equations might help improve the realism of this process in a detailed snowpack model (**?????**). However, the developments of **?** are not sufficiently robust to be available in this official release. More stable and numerically efficient alternatives are emerging and might be preferred in the future following the recommendations of **?**.

2.4.19 Scavenging of LAP

When LAP are activated (Section 2.4.10), liquid water percolation may carry a fraction of LAP mass (**?**):

$$\mathcal{M}_{i,j}^{+} = \mathcal{M}_{i,j}^{-} - \mathcal{F}_i \times C_{\text{scav},j} \times \frac{\mathcal{M}_{i,j}^{-}}{m_i} \tag{136}$$

The scavenging coefficient $C_{\text{scav},j}$ can be adjusted by the user for each LAP j (default values are set to $C_{\text{scav},j} = 0$ i.e. no scavenging).

2.4.20 Sublimation and deposition

Sublimation and deposition are accounted for ~~adding or deleting~~ by adding or removing mass to the surface layer accordingly with the surface latent heat flux (Eq. 85 where T_1 is the solution of Eq. 107, 109 or 111 depending on the occurrence of melting). The corresponding evolution of the surface layer properties is described by Eq. 137-138.

$$z_{i1}{}^{+} = \max(z_{i1} + LE \frac{\Delta t}{L_s(\rho_i^- - \frac{l_i}{z_i^-})} \frac{\Delta t}{L_s(\rho_1^- - \frac{l_1}{z_1^-})}, 0) \tag{137}$$

$$\rho_{i1}{}^{+} = \rho_{i1}{}^{-} + \frac{l_i}{z_i^+} \frac{l_1}{z_1^+} \tag{138}$$

Microstructure properties are not modified. Therefore, in case of deposition, the current Crocus snow model only increases the mass of the surface snow layer but it does not allow ~~to follow~~ the formation of a dedicated layer with microstructure properties typical of surface hoar. As a result, it is neither possible to track the burying of surface hoar ~~as a dedicated snow layer~~in the snowpack. In the very unusual cases when the mass of the surface layer is insufficient for sublimation ($-LE\frac{\Delta t}{L_s} > z_i(\rho_i^- - \frac{l_i}{z_i^-})$ which only occurs for extremely thin snowpacks), the quantity $-LE\frac{\Delta t}{L_s} - z_i(\rho_i^- - \frac{l_i}{z_i^-})$ is later extracted from the first soil layer in the ISBA-DIF soil scheme to conserve energy, and the remaining liquid water, if any, is transferred to the next layer. A homogeneous regridding is applied before the next time step.

2.4.21 Unloading from vegetation

In case of unloading from vegetation, the initial implementation of MEB consisted in adding the unloaded mass to the mass of solid precipitation m_{SP}, following Eq. 36 and 37 for snow density and microstructure. This could lead to unrealistic surface density when unloading occurred in cold conditions. A recent new parameterization instead attributes a fixed density ρ_{UN} and microstructure properties d_{UN} and S_{UN} to unloaded snow. The associated snow mass is either aggregated to the surface snow layer following Eq. 12-20 or associated to a new snow layer depending on the properties of the surface snow layer.

3 Diagnoses

This section describes the complementary diagnoses of a Crocus simulation provided in addition to the state variables of the model. These diagnoses can be useful for model evaluation or interpretation in research or operational applications.

3.1 Diagnoses of recent, wet or refrozen snow

We define n_X as the number of snow layers more recent than X days, i.e. satisfying the following condition:

$$A_i < X\,\forall i \in [1, n_X] \tag{139}$$

Thus, thickness Z_X and mass M_X of snow more recent than X days are defined by:

$$Z_X = \sum_{i=1}^{n_X} z_i \tag{140}$$

$$M_X = \sum_{i=1}^{n_X} m_i \tag{141}$$

These diagnoses are provided for $X \in [0.5, 1, 3, 5, 7]$. Similarly, the number of wet and refrozen snow layers from the surface n_w and n_r are defined by:

$$l_i > 0\,\forall i \in [1, n_w] \tag{142}$$

$$l_i < 0\,\forall i \in [1, n_r] \tag{143}$$

The thickness of wet and refrozen snow Z_w and Z_r are diagnosed by:

$$Z_w = \sum_{i=1}^{n_w} z_i \tag{144}$$

$$Z_r = \sum_{i=1}^{n_r} z_i \tag{145}$$

3.2 Grain type classification

For each snow layer, a diagnosis of grain type $\cancel{\Theta_i}$ Ψ_i is derived from values of optical diameter and sphericity through the following classification (Eq. 146) in which $\delta_i(d_i, S_i)$ and $g_{s_i}(d_i, S_i)$ are defined respectively by Eq. D2 and D6. The values taken by $\cancel{\Theta_i}$ Ψ_i are taken from the International Snow Classification (**?**) and include PP (Precipitation Particles), DF (Decomposited Fragments), RG (Rounded Grains), FC (Faceted Crystals), DH (Depth Hoars), MF (Melt Forms) and combinations of these types.

$$
\begin{cases}
\text{If } d_i < 10^{-4}(4 - S_i) &
\begin{cases}
\text{If } \delta_i \in [0;0.3[&
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{DF+FC} \\
\text{else} & \Psi_i = \text{DF+RG}
\end{cases} \\
\text{If } \delta_i \in [0.3;0.6[& \Theta\Psi_i = \text{DF} \\
\text{If } \delta_i \in [0.6;0.8[& \Theta\Psi_i = \text{PP+DF} \\
\text{If } \delta_i \in [0.8;1.0] & \Theta\Psi_i = \text{PP}
\end{cases} \\
\text{else} &
\begin{cases}
\text{If } h_i = 0 &
\begin{cases}
\text{If } S_i \in [0;0.2[& \Psi_i = \text{FC} \\
\text{If } S_i \in [0.2;0.8[& \Psi_i = \text{RG+FC} \\
\text{If } S_i \in [0.8;1.0] & \Psi_i = \text{RG}
\end{cases} \\
\text{If } h_i = 1 &
\begin{cases}
\text{If } g_{s_i} \in [0,0.55[&
\begin{cases}
\text{If } S_i \in [0;0.2[& \Psi_i = \text{FC} \\
\text{If } S_i \in [0.2;0.8[& \Psi_i = \text{RG+FC} \\
\text{If } S_i \in [0.8;1.0] & \Psi_i = \text{RG}
\end{cases} \\
\text{If } g_{s_i} \in [0.55,1.05[&
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{FC+DH} \\
\text{else} & \Psi_i = \text{MF+DH}
\end{cases} \\
\text{If } g_{s_i} \geq 1.05 &
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{DH} \\
\text{else} & \Psi_i = \text{MF+DH}
\end{cases}
\end{cases} \\
\text{If } h_i \in (2;4) &
\begin{cases}
\text{If } g_{s_i} \in [0,0.55[&
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{MF+FC} \\
\text{else} & \Psi_i = \text{RG+MF}
\end{cases} \\
\text{If } g_{s_i} \geq 0.55 &
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{MF+FC} \\
\text{else} & \Psi_i = \text{MF}
\end{cases}
\end{cases} \\
\text{If } h_i \in (3;5) &
\begin{cases}
\text{If } S_i < 0.55 & \Psi_i = \text{MF+FC} \\
\text{else} & \Psi_i = \text{MF+DH}
\end{cases}
\end{cases}
\end{cases}
\tag{146}
$$

3.3 Snowmaking diagnoses

The water use for snowmaking is obtained by:

$$
\mathcal{V}_{\mathrm{SM}} = \frac{m_{\mathrm{SM}} \mathcal{A}_{\mathrm{SM}}}{(1 - \mathcal{L}_{\mathrm{SM}})\rho_w}
\tag{147}
$$

It is cumulated since the beginning of the season in the model diagnoses.

3.4 Optical diagnoses

The specific surface area SSA_i $(\mathrm{m}^2\mathrm{kg}^{-1})$ of each snow layer i is directly diagnosed from the optical diameter by:

$$\mathrm{SSA}_i = \frac{6}{d_i \rho_I} \tag{148}$$

When CSNOWRAD=T17, spectral albedo values are also provided from an integration of incoming and absorbed radiations over user-defined spectral bands.

3.5 Mechanical diagnoses

3.5.1 Penetration resistance

A penetration resistance is computed for each layer. It is designed to represent the measurement obtained by the rammsonde commonly used in field snowpack observations (**?**). The value of penetration resistance $\mathcal{R}_{p_i}$ is inferred for each layer $i \in [1, N]$ by Eq. 149 from microstructure properties (d_i, S_i and the transformation $g_{s_i}(d_i, S_i)$ following Eq. D6), density ρ_i, liquid water content w_i $(\mathrm{kg\,m^{-3}})$ and temperature θ_i $(^\circ\mathrm{C})$. The result is expressed in kgf $(1\,\mathrm{kgf} \simeq 9.81\,\mathrm{N})$.

$$\begin{cases} \text{If } d_i < 10^{-4}(4 - S_i): & \mathcal{R}_{p_i} = \frac{d_i \times 10^4 - 4 + S_i}{S_i - 3} \times \max(1, 0.018\rho_i - 1.363) + \frac{1 - d_i \times 10^4}{S_i - 3} \times \max\left[2, S_i(0.17\rho_i - 31) + (1 - S_i)(0.085\rho_i - 14.9)\right] \\ \text{else:} & \begin{cases} \text{If } \Psi_i = \text{RG}: & \begin{cases} \text{If } \rho_i < 200: \mathcal{R}_{p_i} = 3 \\ \text{else:} \mathcal{R}_{p_i} = 0.17\rho_i - 31 \end{cases} \\ \text{If } \cancel{\Theta}\Psi_i = \text{RG+FC}: & \begin{cases} \text{If } \rho_i < 200: \mathcal{R}_{p_i} = 2 \\ \text{else:} \mathcal{R}_{p_i} = S_i \times (0.17\rho_i - 31) + (1 - S_i) \times (0.17\rho_i - 31) \cdot (0.8 - g_{s_i}(d_i, S_i)) + 2 \times g_{s_i}(d_i, S_i) \end{cases} \\ \text{If } \cancel{\Theta}\Psi_i \in \begin{bmatrix} \text{FC} \\ \text{FC+DH} \end{bmatrix}: & \begin{cases} \text{If } d_i > 4 \times 10^{-4}(S_i + 1),\ \mathcal{R}_{p_i} = 2 \\ \text{else} \begin{cases} \text{If } \rho_i < 200: \ \mathcal{R}_{p_i} = 3 \cdot (0.8 - g_{s_i}(d_i, S_i)) + 2 \cdot g_{s_i}(d_i, S_i) \\ \text{else: } \ \mathcal{R}_{p_i} = 0.17\rho_i - 31 \end{cases} \end{cases} \\ \text{If } \cancel{\Theta}\Psi_i \in \begin{bmatrix} \text{MF} \\ \text{MF+RG} \\ \text{MF+FC} \\ \text{MF+DH} \end{bmatrix}: & \begin{cases} \text{If } (\theta_i < -0.2 \text{ or } w_i < 5): \ \mathcal{R}_{p_i} = \max(10, 0.103\rho_i - 19.666) \\ \text{else} \begin{cases} \rho_i < 250,\ \mathcal{R}_{p_i} = 1 \\ 250 \le \rho_i < 350,\ \mathcal{R}_{p_i} = 2 \\ \rho_i \ge 350,\ \mathcal{R}_{p_i} = 0.16\rho_i - 54 \end{cases} \end{cases} \\ \text{If } \cancel{\Theta}\Psi_i = \text{DH}: & \mathcal{R}_{p_i} = 2 \end{cases} \end{cases} \tag{149}$$

3.5.2 Shear strength

To be able to compute stability ~~indices~~ indexes of the snowpack, a shear resistance $\mathcal{R}_{s_i}$ ~~computed~~ is diagnosed for each layer i. It is also computed from microstructure properties, density, liquid water content and temperature through Eq. 150. $\mathcal{R}_{s_i}$ is expressed in $\mathrm{kgf\,dm^{-2}}$ ($1\,\mathrm{kgf\,dm^{-2}} \simeq 0.981\,\mathrm{kPa}$).

$$\begin{cases} \left[\begin{array}{l} \text{If } w_i \geq 5 \\ \text{and } \Psi_i \in \left\{ \begin{array}{l} \text{RG, MF, RG+FC,} \\ \text{RG+MF,DF+RG,} \\ \text{MF+FC, MF+DH} \end{array} \right\} \end{array} \right] & \begin{cases} \text{If } \rho_i < 200: & \mathcal{R}_{s_i} = 0.1 \\ \text{If } 200 \leq \rho_i < 320: & \mathcal{R}_{s_i} = 0.02\rho_i - 3.9 \\ \text{If } \rho_i \geq 320: & \mathcal{R}_{s_i} = 0.068\rho_i - 18.64 \end{cases} \\ \text{else:} & \mathcal{R}_{s_i} = \max(0.05, \mathcal{C}_1(S_i,h_i) \times \mathcal{C}_2(d_i,S_i) \times \mathcal{C}_3(d_i,S_i,h_i) \times \mathcal{C}_4(w_i,\rho_i) \\ & \times \mathcal{C}_5\,(d_i,S_i,h_i,w_i,\rho_i) \times (\rho_i^2 \cdot 10^{-4} - 0.6) + 0.12) \end{cases} \tag{150}$$

$\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4$ and $\mathcal{C}_5$ multiplicative functions are defined in Appendix G.

3.6 MEPRA

MEPRA was a standalone module designed to estimate the avalanche hazard from the snowpack stratigraphy simulated by Crocus, from mechanical diagnosis and expert rules (**?**). It has been fully implemented in the SURFEX platform, and its output are now available with the other diagnostic variables. The general idea is to compare the shear strength $\mathcal{R}_{s_i}$ as defined previously, to the shear stress in the layer. For natural release, only the weight of overlying layers is taken into account while the load related to the presence of a skier at the snowpack surface is added to represent the accidental triggering. Expert rules are then defined to determine a hazard ~~indice~~ index from these mechanical stability indicators, both for natural release and accidental triggering. ~~Expert rules were defined with the work of~~ These expert rules were initially implemented by **?** but remained largely unpublished and ~~evolved through versions of Crocus, mainly from~~ have evolved to answer to feedbacks of operational ~~forecasters~~avalanche forecasters, without detailed documentation. Equations implemented in the current version of SURFEX are described below. They are only valid for slope angle ~~valus $\gamma = 40^o$~~ $\Theta = 40^o$.

3.6.1 Mechanical stability of snowpack layers

A simple mechanical diagnosis for stability is computed by dividing the shear resistance $\mathcal{R}_{s_i}$ by the shear in the layer. Two values are computed, to discriminate between natural avalanche activity and human triggering.

Natural release

The stability ~~indice~~ index $\mathcal{S}_{\mathrm{nat}_i}$ for natural release is defined in each layer i as follows:

$$\mathcal{S}_{\mathrm{nat}_i} = \frac{\mathcal{R}_{s_i}}{\sigma_{i+1}} \tag{151}$$

where σ_{i+1} is the weight of overlying layers (including considered layers), projected on the slope-parallel axis, as defined by Eq. 54.

Accidental triggering

The stability ~~indice~~ index for accidental triggering $\mathcal{S}_{\mathrm{acc}_i}$ is similar to equation 151, with a supplementary term for shear stress to represent the additional load on the snowpack:

$$\mathcal{S}_{\mathrm{acc}_i} = \frac{\mathcal{R}_{s_i}}{\sigma_{i+1} + \Xi_i \sigma_{\mathrm{acc}_i}} \tag{152}$$

where σ_{acc_i} is designed to represent the shear load induced by a skier, defined as a piecewise linear decreasing function of depth $Z_i = \sum_{j=1}^{i} z_j$:

$$\sigma_{\mathrm{acc}_i} = \begin{cases} 4 - 15Z_i & \text{if } Z_i < 0.1 \\ 2.5 - 10(Z_i - 0.1) & \text{if } 0.1 \leq Z_i < 0.15 \\ 2 - 8(Z_i - 0.15) & \text{if } 0.15 \leq Z_i < 0.2 \\ 1.6 - 4(Z_i - 0.2) & \text{if } 0.2 \leq Z_i < 0.35 \\ 1 - 2(Z_i - 0.35) & \text{if } 0.35 \leq Z_i < 0.5 \\ 0.7 - 1.5(Z_i - 0.5) & \text{if } 0.5 \leq Z_i < 0.8 \\ 0 & \text{if } 0.8 \leq Z_i \end{cases} \tag{153}$$

and Ξ_i is designed to represent the bonding effect reducing the shear stress in the layers, defined by:

$$\Xi_i = \frac{\sum_{j=1}^{i} m_j \xi_j}{\sum_{j=1}^{i} m_j} \text{ where } \xi_i : \begin{cases} \text{If } \Psi_i \in \begin{Bmatrix} \text{MF, RG+MF,} \\ \text{MF+FC, MF+DF} \end{Bmatrix} & \begin{cases} \text{If } \theta_i < -0.2,\ \xi_i = 0.5 \\ \text{else: } \xi_i = 1.1 \end{cases} \\ \text{else:} & \begin{cases} \text{If } \mathcal{R}_{s_i} > 1.5:\ \xi_i = 1.0 \\ \text{else: } \xi_i = 1.2 \end{cases} \end{cases} \tag{154}$$

3.6.2 Hazard ~~indices~~indexes

MEPRA analyses the profiles of $\mathcal{S}_{\mathrm{nat}_i}$ and $\mathcal{S}_{\mathrm{acc}_i}$ mechanical ~~indices~~ indexes with other parameters (mainly grain type, temperature and liquid water content and snow heights), to assess a natural avalanche hazard ~~indice~~ index $\mathcal{H}_{\mathrm{nat}}$, on a scale of 0-5 and accidental hazard ~~indice~~ index $\mathcal{H}_{\mathrm{acc}}$ on a scale 0-3, with a set of expert rules presented below. These hazard ~~indices~~ indexes are associated with levels of instability (one for high instability, Z_h, and one for moderate instability, Z_m, at most). For natural release, a classification between 5 avalanche types is also provided.

Natural release

The natural release analysis relies on a classification of the upper layers of the snowpack (referred as "superior profile") in 4 classes : NEW (new snow), WET (wet snowpack), FRO (refrozen snowpack) or NAN (when it could not be classified in other class), based on the conditions listed in Table **??**. A height of this superior profile Z_{SUP} is also defined below which the snowpack is significantly different, and called inferior profile. The latter is also classified into three classes ~~(SOF , HAR and NAN)~~ : SOF (soft), HAR (hard) and NAN (undefined) as a function of the maximum penetration resistance, following Table **??**.

Levels of high and moderate instabilities are looked for in the superior profile, as the uppermost buried layer where $\mathcal{S}_{\mathrm{nat}_i}$ is below a threshold $\mathcal{S}_h = 2$ or $\mathcal{S}_m = 3$ respectively (or $\mathcal{S}_m = 3.05$ for a NEW type of superior profile):

$$\text{If } \exists j | Z_j > 0.1 \text{ and } Z_j \leq Z_{\text{SUP}} \text{ and } \mathcal{S}_{\text{nat}_j} \leq \mathcal{S}_h : Z_h = Z_j - \frac{z_j}{2} \tag{155}$$

$$\text{If } \exists k | Z_k > 0.1 \text{ and } Z_j \leq Z_{\text{SUP}} \text{ and } \mathcal{S}_{\text{nat}_j} \leq \mathcal{S}_m \text{ and } j \neq k : Z_m = Z_k - \frac{z_k}{2} \tag{156}$$

A first natural avalanche hazard ~~indice~~ index $\mathcal{H}_{\text{nat}}$ is defined depending on depths of instability level Z_h and Z_m, superior profile height Z_{SUP} and superior profile type with the following expert rules:

$$\mathcal{H}_{\text{nat}_1} = \begin{cases} Z_h > 0.8 \begin{cases} \text{NEW} \rightarrow 5 \\ \text{WET} \rightarrow 5 \\ \text{FRO} \rightarrow 3 \end{cases} \\ Z_h > 0.4 \begin{cases} \text{NEW} \rightarrow 4 \\ \text{WET} \rightarrow 4 \\ \text{FRO} \rightarrow 3 \end{cases} \\ Z_h > 0.2 \begin{cases} \text{NEW} \rightarrow 2 \\ \text{WET} \rightarrow 2 \\ \text{FRO} \rightarrow 3 \end{cases} \\ Z_h > 0 \begin{cases} \text{NEW} \rightarrow 1 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 1 \end{cases} \\ Z_h \text{ not defined} \rightarrow 0 \end{cases} \quad \mathcal{H}_{\text{nat}_2} = \begin{cases} Z_m > 0.8 \begin{cases} \text{NEW} \rightarrow 3 \\ \text{WET} \rightarrow 3 \\ \text{FRO} \rightarrow 1 \end{cases} \\ Z_m > 0.4 \begin{cases} \text{NEW} \rightarrow 3 \\ \text{WET} \rightarrow 3 \\ \text{FRO} \rightarrow 1 \end{cases} \\ Z_m > 0.2 \begin{cases} \text{NEW} \rightarrow 3 \\ \text{WET} \rightarrow 3 \\ \text{FRO} \rightarrow 1 \end{cases} \\ Z_m > 0 \begin{cases} \text{NEW} \rightarrow 1 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 1 \end{cases} \\ Z_m \text{ not defined} \rightarrow 0 \end{cases} \quad \mathcal{H}_{\text{nat}_3} = \begin{cases} Z_{\text{SUP}} > 0.8 \begin{cases} \text{NEW} \rightarrow 1 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 0 \end{cases} \\ Z_{\text{SUP}} > 0.4 \begin{cases} \text{NEW} \rightarrow 1 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 0 \end{cases} \\ Z_{\text{SUP}} > 0.2 \begin{cases} \text{NEW} \rightarrow 1 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 0 \end{cases} \\ Z_{\text{SUP}} > 0 \begin{cases} \text{NEW} \rightarrow 0 \\ \text{WET} \rightarrow 1 \\ \text{FRO} \rightarrow 0 \end{cases} \\ Z_{\text{SUP}} \text{ not defined} \rightarrow 0 \end{cases} \tag{157}$$

$$\text{NEW, WET, or FRO} \rightarrow \mathcal{H}_{\text{nat}} = \begin{cases} 2 & \text{if } \mathcal{H}_{\text{nat}_1} = 2 \\ \max(\mathcal{H}_{\text{nat}_1}, \mathcal{H}_{\text{nat}_2}, \mathcal{H}_{\text{nat}_3}) & \text{otherwise} \end{cases}$$

$$\text{NAN} \rightarrow \mathcal{H}_{\text{nat}} = 0$$

The avalanche situation is then classified in 6 classes following Appendix **??**. Finally, $\mathcal{H}_{\text{nat}}$ is updated by expert rules accounting from the temporal evolution between times $t - \Delta t_M$ and t following Appendix **??**. Note that these rules are sensitive to the MEPRA time step Δt_M, set to 3 hours by **?**.

Accidental triggering

$\mathcal{H}_{\text{acc}}$ is based on the identification of a slab structure in the snowpack including a slab (layer i and possibly layers above) over a weak layer (layer $i+1$). This structure is identified through the following conditions:

$$\begin{cases} \text{MF} \notin \Psi_{i-1} \\ \mathcal{R}_{s_i} > 1.3 \\ \Psi_i \in [\text{DF+RG, RG, RG+FC, DF}] \\ 0.01m \leq Z_i < 1m \\ \Psi_{i+1} \in [\text{FC, DH, FC+DH, PP, PP+DF, DF}] \end{cases} \tag{158}$$

Similarly to the natural instability levels (Eq. 155 and 156), levels of high accidental instability and moderate accidental instability are looked for among the identified weak layers satisfying Eq. 158:

$$\text{If } \exists i | \Theta\Psi_{i+1} \in [\text{FC, DH, FC+DH}] \text{ and } \mathcal{S}_{\text{acc}_{i+1}} < 1.5 : Z_h = Z_{i+1} - \frac{z_{i+1}}{2} \tag{159}$$

$$\text{If } \exists i | \Theta\Psi_{i+1} \in [\text{PP, PP+DF, DF}] \text{ or } \left(\Theta\Psi_{i+1} \in [\text{FC, DH, FC+DH}] \text{ and } 1.5 \leq \mathcal{S}_{\text{acc}_{i+1}} < 2.5 \right) : Z_m = Z_{i+1} - \frac{z_{i+1}}{2} \tag{160}$$

The accidental hazard ~~indice~~ index is finally defined with the following expert rules:

$$\begin{cases} \text{If } \exists i | \text{ Eq. 159} : \begin{cases} \text{If } Z_c < 0.01 : \mathcal{H}_{\text{acc}} = \max(3, \mathcal{H}_{\text{eq}}) \\ \text{else:} \mathcal{H}_{\text{acc}} = \max(1, \mathcal{H}_{\text{eq}}) \end{cases} \\ \text{If } \exists i | \text{ Eq. 160} : \begin{cases} \text{If } Z_c < 0.01 : \mathcal{H}_{\text{acc}} = \max(2, \mathcal{H}_{\text{eq}}) \\ \text{else:} \mathcal{H}_{\text{acc}} = \max(1, \mathcal{H}_{\text{eq}}) \end{cases} \\ \text{If } \nexists i | \text{Eq. 159 or Eq. 160} : \mathcal{H}_{\text{acc}} = \max(0, \mathcal{H}_{\text{eq}}) \end{cases} \tag{161}$$

where $\mathcal{H}_{\text{eq}}$ is a function of the natural ~~indice~~ index $\mathcal{H}_{\text{nat}}$ following Appendix **??** and Z_c is the cumulated depth of crusts above a layer:

$$Z_{c_h} = \sum_i z_i \text{ for } i | \theta_i < 0.2 \text{ and MF} \in \Theta\Psi_i \text{ and } Z_i < Z_h \text{ and } (\text{PP} \notin \Theta\Psi_j \text{ and DF} \notin \Theta\Psi_j \; \forall j \in [1, i-1]) \tag{162}$$

$$Z_{c_m} = \sum_i z_i \text{ for } i | \theta_i < 0.2 \text{ and MF} \in \Theta\Psi_i \text{ and } Z_i < Z_m \text{ and } (\text{PP} \notin \Theta\Psi_j \text{ and DF} \notin \Theta\Psi_j \; \forall j \in [1, i-1]) \tag{163}$$

Note that only one value for Z_h and Z_m is provided in the diagnoses output file. If $\mathcal{H}_{\text{acc}} = \mathcal{H}_{\text{eq}}$ in Eq. 161, they correspond to Eq. 155 and 156, otherwise they correspond to Eq. 159 and 160.

4 Technical features

4.1 Implemented simulation geometries

All variables in the code are defined as vectors including at least a spatial dimension, and when necessary the snow layers as second dimension. The implication in terms of numerical efficiency of loops in the code is described in detail in Appendix **??**. This 1D spatial dimension allows to use either discontinuous collections of points or regular grids for simulations. A typical example of a collection of points is the semi-distributed geometry based on homogeneous massifs and topographic classes which has been used for more than 30 years for operational simulations in French mountains and in the associated 66-year reanalysis (**?**). Gridded experiments can also be easily defined through the standard SURFEX tools which include regular latitude-longitude coordinates or various conformal projections. Over the French territory, the Lambert 93 projection is recommended as a national standard for gridded simulations (i.e. **??**).

4.2 Parallelization and numerical efficiency

In the general case where Crocus is not coupled with a snow transport model, the processes currently represented by the model do not involve any mass or energy exchange between the snowpacks of the different spatial units. Therefore, parallelization can be very efficiently applied without any need of communication between processors in the physical model. A distributed memory parallelism has be chosen for that purpose in the offline driver of the SURFEX platform based on MPI libraries. However, the standard SURFEX Input-Output routines are not currently parallelized. Therefore, all input and output data are forwarded to a single processor. As a result, over large domains, the efficiency of parallel computing is currently limited by the saturation of the IO processor, in all cases but especially when the user chooses to output a large number of diagnoses at high temporal resolution. The XIOS library (**?**) implemented in SURFEX is designed to deal with this issue but the implementation of its compatibility with Crocus variables is still in progress. The numerical cost of the model itself depends on the number of layers created by the model. Therefore, it can ~~significantly vary from a~~ vary significantly from one domain to another and from a season to another, depending on the number of layers of each simulated snowpack which highly depends on total snow depth.

To give the magnitude of the numerical cost, Table 3 presents the computing time of a 1-year simulation over 4471 simulation points in the French Alps. It must be noticed ~~than~~ that the numerical cost of Crocus is very low compared to the cost of IO, providing a clear guidance for priorities in future optimizations. Crocus also only represents 26% of the computation time of the ISBA land surface model itself. Even in an alpine region, the variability of snow cover in time and space makes the snow component relatively cheap with a similar level of complexity compared to the 20-layer soil model running all year over all points. As a result, the numerical cost of Crocus (0.3 $\mathrm{cores} \times \mathrm{s}$ per year and simulation point for this example) can not be considered as a valid argument to prefer simpler models in large scale applications of ~~LSM~~ Land Surface Models or coupled applications.

Considering the partitioning of numerical costs between the different subroutines within Crocus, it appears clearly that the complexity of the discretization rules emphasized by Eq. 11 has a significant impact as this routine represents 33% of the whole model cost. The metamorphism routine is the second most contributing routine although its cost has been considerably reduced compared to previous versions (**?**). The numerical core of the model solving heat diffusion and energy balance is very efficient contributing to less than 6% of the cost. Numerical optimizations are still possible in some routines representing an unjustified contribution compared to their low complexity (e.g. thermal conductivity). Possible optimizations may concern the management of loops (cf. Appendix **??**) or some iterative calls to scalar functions in external modules.

The memory consumption of the model is relatively low with the standard physical options. In the experiment described above, the maximum Resident Set Size (RSS) is lower than 17 GB. However, the high spectral resolution of the TARTES optical scheme slightly increases the memory consumption (25 GB) and highly increases the total numerical cost when this option is activated (28588 $\mathrm{cores} \times \mathrm{s}$ for the TARTES module in an experiment identical to Table 3 but with SNOWRAD=T17, i.e. about 20 times the reference numerical cost of Crocus).

Table 3. Numerical cost of a 1-year simulation from 1 Aug 2023 to 1 Aug 2024 over 4471 simulation points corresponding to the French Alps domain described in **?** and the default physical options as in **?**. The code was compiled with Intel®MPI library version 2018.5.274 and O2 optimization level. Output are minimized (reduced to daily snow depth), otherwise the total execution time would be highly increased. The simulation is performed using 80 MPI threads on 80 physical cores on one 2.2 GHz AMD©Rome computing node constituted by 2 sockets of 64 cores. Results are presented in cores $\times$ s. The real elapsed time for this simulation in this architecture is 449 s.

	Time (cores $\times$ s)	Ratio with Crocus time (%)
SURFEX run	35964	2480
IO reading and communications	29390	2026
ISBA land surface model	5513	380
Crocus snow model	1450	100
Snowfall and vertical discretization	491	33.4
Metamorphism	173	11.9
Absorption of solar radiation	162	11.2
Compaction	95	6.6
Regridding (aggregation / dissociation)	77	5.3
Heat diffusion (Eq. 107-108)	64	4.4
Thermal conductivity	61	4.2
Percolation and refreezing	58	4.0
Melting	30	2.1
Aggregation of vanishing layers	27	1.9
Drift	25	1.7
Diagnostic of energy fluxes	23	1.6
Energy balance (Eq. 102-106)	20	1.4

4.3 Running environment and visualization

Beyond the FORTRAN code itself, most offline applications of the model have to deal with the management of input and output files to perform various experiments with different forcing files, namelists, or binaries, different setting of initial and final simulation dates, standard initialization procedures (soil spinup, etc.). Therefore, a common running environment of the model in Python is provided in an independent package called snowtools ~~coming with a~~ with full user documentation and an interface for technical support (cf. Code availability section).

Crocus scalar diagnoses can be easily processed by any scientific plotting software supporting netcdf format as input. However, the irregular vertical discretization of the snowpack model complexifies the visualization of the simulated profiles. A simple software provided in the snowtools package is able to combine the variables to plot with the depth of each layer to produce detailed instantaneous stratigraphies or temporal evolution of a given stratified variable ~~. For spatialized~~ (as in Appendix **??**). For spatial simulations, plotting the spatial variability of the vertical structure of the snowpack is still a challenge.

4.4 Externalization

Although the reference implementation of Crocus is within the SURFEX land surface modelling platform (**?**), there is an increasing need of being able to couple Crocus with other land surface schemes. For that purpose, an externalized version of the source codes is now available as an independent Fortran library that can be compiled alone and called by other land surface models (cf. Code availability section). It includes all the processes described in this paper except the coupling with external components (snow transport modules and MEB vegetation module). Thus, Crocus is also now fully integrated within the SVS2 land surface system (**?**). It was also recently implemented within the Flexible Snow Model (FSM2, **?**) allowing the coupling with its more detailed vegetation model (**?**). SURFEX, SVS2 and FSM2 rely on a unique code repository of Crocus that guarantees the long-term maintenance and convergence of the code and therefore facilitates the contributions of different research groups to the model developments. For instance, as mentioned for several processes in Section 2, various new parameterizations better suited for Arctic snow were recently proposed within the SVS2 implementation (**??**). Thanks to this method, these developments will integrate soon a future release and be beneficial for SURFEX and FSM2 applications. Therefore, we strongly encourage other groups that have copied the code within their specific applications to try to converge towards this unique code version. This includes the implementation of Crocus within WRF-Hydro (**?**), the coupling of Crocus with Noah ~~LSM~~ Land Surface Model (**?**), the MAR regional climate model largely used for polar regions (**??**) and from which the Crocus version has diverged for a long time (**?**), and even applications which have only extracted specific routines such as the CryoGrid permafrost model (**?**).

5 Review of evaluations and scientific applications

5.1 With a local scale meteorological forcing

The most direct evaluations of the model are performed on well-instrumented sites allowing to minimize errors in the meteorological forcing and in the observations used for evaluation. Model skills have been documented in detail at the Col de Porte experimental site (**??**), a mid-elevation meadow in French Alps. The very first evaluations of **?** present qualitative evaluations of surface and internal temperature, snow depth, and basal runoff during short periods of the winter 1986-1987. **?** extended the evaluations to the whole winter 1988-1989 on the same variables as well as a subjective comparison between the simulated stratigraphies and weekly observed profiles. Extensions of the evaluation period were successively published by **?????** in the context of model intercomparisons, and also by **?**. These papers also included evaluations of albedo and ~~SWE~~Snow Water Equivalent. **?** extended these evaluations to all multiphysics options and accounted for observation uncertainties usually ignored in previous papers. The ~~scores~~ accuracy of the model on these variables ~~have~~ has not significantly changed since the first years of development. Complementary evaluations of density and microstructure profiles are provided by **?** and **?**. However, quantitative evaluation of internal snow properties is still a methodological challenge due to frequent ~~discreapancies~~ discrepancies between numerical and observed snow layers that can easily lead to double penalty issues. This can only be partly solved with vertical adjustments algorithms (**??**) or with an expert layer tracking (**?**).

More challenging evaluations include a variety of environmental and climate conditions. Evaluations driven by local meteorological observations were performed in Svalbard (**??**), at the high elevation site of Weissflujoch, Switzerland (**?**), over tropical glaciers and moraines in Bolivia (**?**) and Ecuador (**?**), at Sherbrooke University, Quebec (**?**), at Torgnon, Italy (**?**). The most comprehensive evaluations in terms of number of sites and years were performed on 10 contrasted sites through the Earth System Model-Snow Model Intercomparison Project (ESM-SnowMIP, **??**) and with a few more sites for albedo evaluations by **?**. Typical errors on ~~SWE, density~~snow depth, SWE, albedo, surface temperature and ground temperature ~~are~~ were in the range of state-of-the art snow models ~~but~~ as illustrated with up-to-date simulations with this model version in Appendix **??**. A similar overall skill can be obtained by simpler models on these variables. ~~A~~ The cold bias in surface temperature ~~was identified in these experiments (**?**). It~~ identified by **?** may be attributed to the parameterization of turbulent fluxes which are suspected to be underestimated ~~in the standard option~~ with the previously standard configuration ($\mathrm{Ri}_l = 0.2$) but this bias ~~can be~~ is removed with the ~~other options (**??**)~~new default value ($\mathrm{Ri}_l = 0.026$, **??**). This assumption was also recently supported by more detailed evaluations of all components of the energy balance including eddy-covariance observations of turbulent fluxes in Québec (**?**) and in Finnish peatlands (**?**). However, it is especially important to be aware ~~that the equifinality of~~ of the equifinality between the different empirical parameterizations ~~and~~ (i.e. different model configurations providing similar model results) and to be cautious with the complex compromises in multivariate evaluations ~~(**??**)~~ (the optimal model configuration can change depending on the evaluated variable, **??**). These difficulties should encourage future attempts to improve processes representations to be tested robustly with ensemble multiphysics simulations and multivariate evaluations (e.g. **?**).

5.2 With a regional scale meteorological forcing

In many other applications, the model was forced by meteorological reanalyses (e.g. **???**) or short-term forecasts (e.g. **??**) or a combination of both (**?**). Although these studies often include snow depth evaluations on a large range of stations, in this case the resulting modelling errors of any snow model are dominated by errors in the meteorological forcing (**??**) and Crocus does not make an exception (**????**). Thus, the attribution of some limitations of the simulation results to the snowpack model itself is difficult. However, this is sometimes the only possible method for the assessment of some specific processes. For instance, the simulated concentrations of Light-Absorbing Particles (**?**), the spectral reflectances from the TARTES optical scheme (**?**), the blowing snow fluxes (**??**) were only evaluated in such context. The same applies to the ability of the model to reproduce the properties of polar snow (e.g. **????**), the spatial distribution of snow conditions depending on topography (**???**), or ~~its adequation with~~ complex remote sensing signals (**?**). Similar simulation frameworks are used to investigate the suitability of the model for hydrological diagnoses (e.g. **?**), glacier mass balance (e.g. **??**) or avalanche activity (**??**).

Based on the confidence provided by these available evaluations, the model is used for various purposes including the understanding of internal physical processes (e.g. **???**), the quantification of contributions to the energy balance (e.g. **???**, for light-absorbing particles), the monitoring of the long-term climatology of extreme snow loads (**?**) or avalanche activity (**??**), the investigation of the links between snow cover and alpine ecosystems evolutions (**?**), and climate projections of natural snow conditions (**??**), avalanche hazard (**?**) and ski resorts operating conditions (**???**). The development efforts were originally dedicated to operational avalanche hazard forecasting (**?**). However, after about 30 years of operation, the model did not become the main tool of the forecasters because this application is especially sensitive to uncertainties (**?**) among numerous other challenges as reviewed by **?**. The statistical post-processing of simulations through various techniques of artifical intelligence (**???**) might provide guidance to improve the practical interest of such simulations for operational applications including avalanche forecasting.

5.3 Towards data assimilation

The long memory of the snowpack on the past meteorological conditions and past snow processes imply that all sources of modelling errors tend to cumulate all along the season. Therefore, there is a large avenue for data assimilation algorithms in order to improve the initial states of the simulations (**?**). However, the variable dimension size of the Crocus state vector (due to the variable number of layers) and the high non-linearities in the simulated processes make ~~especially challenging the application to this model~~ the application of a number of data assimilation algorithms to this model challenging. Therefore, most recent efforts intended to apply different variants of the Particle Filter to weight the members of an ensemble of simulations according to their distance to some observations (**????**). The different options of the algorithm are described in **?** and the implemented variables are snow depth and optical reflectance, for which the observation operator is simply the identity function as these variables are direct diagnoses of the model. The assimilation of optical reflectance is however constrained by the retrieval errors of this variable in complex terrain (**?**) that still exceed the requirements for an efficient data assimilation (**?**). Large efforts are planned in a near future to extend these possibilities. For some common satellite observations (e.g. snow cover fraction, wet snow fraction), this will require the development of appropriate observation operators from the simulated state variables.

6 Conclusion

This article provides a comprehensive description of all equations implemented in version 3.0.1 of the Crocus snow model. It gathers the recent developments of the last 13 years in a unique publication: (i) modelling of Light-Absorbing Particles, (ii) coupling with the TARTES optical scheme, (iii) modelling of ice layers due to freezing rain, (iv) coupling with the MEB big-leaf vegetation scheme, (v) snow management practices on ski slopes, (vi) coupling with blowing snow schemes, (vii) multiphysics parameterizations for most processes and (viii) diagnosis of the snowpack mechanical stability. In addition, this article documents a number of equations implemented in previous code versions but never published in the literature. It must be mentioned that during the preparation of this publication, a considerable number of errors or ~~discreapancies~~ discrepancies between the code and previous publications were identified and corrected. This is in full agreement with the main conclusions of **?** suggesting that insufficient model documentation is a key factor for the difficulty to improve snow modelling in the last decades. This comprehensive documentation is expected to help snow scientists to better interpret results based on this model. This is especially important in the context of an in-progress extension of Crocus applications in several land surface schemes. This documentation effort is also expected to help the snow modelling community improve numerical models in the future thanks to an accurate knowledge of the existing parameterizations and numerical difficulties. Despite our best efforts to minimize errors, previous literature and experience suggest that some errors may still remain in this publication. In such a case, corrigenda will be associated to this publication in due course.

Appendix A: Attractor profile in layering

The attractor profile z_i^* used in ~~Equation~~ Eq. 11 depends only on the total snow depth Z and number of layers N. Let's define N^* and Z^* by:

$$\begin{cases} \text{if } Z > 20 \text{ and } N > \frac{N_{\max}}{3} + 2 \begin{cases} N^* = N - \frac{N+N_{\max}}{6} \\ Z^* = 3 \end{cases} \\ \text{else } \begin{cases} N^* = N \\ Z^* = Z \end{cases} \end{cases} \tag{A1}$$

Then, the attractor profile is defined by:

$$\begin{cases} z_1^* = \min(0.01, \frac{Z^*}{N^*}) \\ z_2^* = \min(0.0125, \frac{Z^*}{N^*}) \\ z_3^* = \min(0.03, \frac{Z^*}{N^*}) \text{ if } N^* > 3 \\ z_4^* = \min(0.04, \frac{Z^*}{N^*}) \text{ if } N^* > 4 \\ z_5^* = \min(0.05, \frac{Z^*}{N^*}) \text{ if } N^* > 5 \\ z_6^* = \max(0.07, \min(\frac{Z^*}{N^*}, 0.5)) \text{ if } N^* > 6 \\ z_7^* = \max(0.07, \min(\frac{Z^*}{N^*}, 1.)) \text{ if } N^* > 7 \\ z_8^* = \max(0.07, \min(\frac{Z^*}{N^*}, 2.)) \text{ if } N^* > 8 \\ z_9^* = \max(0.07, \min(\frac{Z^*}{N^*}, 4.)) \text{ if } N^* > 9 \\ z_{10}^* = \max(0.07, \min(\frac{Z^*}{N^*}, 10.)) \text{ if } N^* > 10 \\ z_i^* = \max(0.07, \min(\frac{Z}{N},)) \forall i \in [11, N^*] \text{ if } N^* > 11 \\ z_i^* = (i - N^*)\frac{2(Z-3)}{(N-N^*)(N-N^*+1)} \forall i \in [N^*+1, N] \text{ if } Z > 20 \\ z_N^* = \min(0.02, \frac{Z}{N^*}) \\ \text{if } Z > 3 \text{ and } N* > 10 \begin{cases} z_{N-1}^* = 0.66 z_N^* + 0.34 z_{N-3}^* \\ z_{N-2}^* = 0.34 z_N^* + 0.66 z_{N-3}^* \end{cases} \quad \text{instead of previous definitions} \end{cases} \tag{A2}$$

It extends the definition of **?** in order to converge towards a profile allowing a numerically stable resolution of heat diffusion for thick snowpacks and glacier applications.

Appendix B: Adjustment of wind speed

Several parameterizations of the model are formulated with a wind speed at a specific height corresponding to the experimental conditions but might not correspond to the reference height of the forcing variable. In these cases, the wind speed U_z (ms^{-1}) at height z (m) is adjusted assuming a logarithmic profile in the surface boundary layer based following:

$$U_z = U \frac{\ln\left(\frac{z}{z_0}\right)}{\ln\left(\frac{z_u}{z_0}\right)} \tag{B1}$$

where z_0 is the surface roughness length (m).

Appendix C: Growth of faceted crystals in B21 metamorphism parameterizations

The growth of faceted crystals in B21 parameterization (Eq. 46) is based on cold room experiments from **?**. These functions already published by **?** are ~~reminded~~ repeated here for the comprehensiveness of this paper. f, g, h and Φ are dimensionless functions from 0 to 1 given by:

$$f(\theta_i) = \begin{cases} 0 & \text{if} \quad \theta_i < -40\,^\circ\mathrm{C} \\ 0.011 \times (\theta_i + 40) & \text{if} \quad -40 \leq \theta_i < -22\,^\circ\mathrm{C} \\ 0.2 + 0.05 \times (\theta_i + 22) & \text{if} \quad -22 \leq \theta_i < -6\,^\circ\mathrm{C} \\ 0.7 - 0.05\theta_i & \text{otherwise} \end{cases} \tag{C1}$$

$$h(\rho_i) = \begin{cases} 1. & \text{if} \quad \rho_i < 150 \quad \mathrm{kg\,m^{-3}} \\ 1 - 0.004 \times (\rho_i - 150) & \text{if} \quad 150 < \quad \rho_i < 400 \quad \mathrm{kg\,m^{-3}} \\ 0. & \text{otherwise} \end{cases} \tag{C2}$$

$$g(\mathcal{G}_i) = \begin{cases} 0. & \text{if} \quad \mathcal{G}_i < 15\,\mathrm{K\,m^{-1}} \\ 0.01 \times (\mathcal{G}_i - 15) & \text{if} \quad 15 \leq \mathcal{G}_i < 25\,\mathrm{K\,m^{-1}} \\ 0.1 + 0.037 \times (\mathcal{G}_i - 25) & \text{if} \quad 25 \leq \mathcal{G}_i < 40\,\mathrm{K\,m^{-1}} \\ 0.65 + 0.02 \times (\mathcal{G}_i - 40) & \text{if} \quad 40 \leq \mathcal{G}_i < 50\,\mathrm{K\,m^{-1}} \\ 0.85 + 0.0075 \times (\mathcal{G}_i - 50) & \text{if} \quad 50 \leq \mathcal{G}_i < 70\,\mathrm{K\,m^{-1}} \\ 1. & \text{otherwise} \end{cases} \tag{C3}$$

$$\Phi = 1.0417.10^{-9}\,\mathrm{m\,s^{-1}} \tag{C4}$$

Appendix D: New formalism of metamorphism

The translation of the original metamorphism parameterizations in terms of dendricity, sphericity and size from **?** to the new formalism of **?** in terms of optical diameter and sphericity, was based on the original expression of the optical diameter d_i as a function of dendricity δ_i, sphericity S_i and grain size g_{s_i} (as already published in Eq. 13 of **?**):

$$d_i = \begin{cases} 10^{-4}\left[\delta_i + (1-\delta_i)(4-S_i)\right] & \text{if } \delta_i > 0 \text{ (dendritic case)} \\ g_{s_i} \times S_i + (1-S_i) \times \max\left(4.10^{-4}, \frac{g_{s_i}}{2}\right) & \text{if } \delta_i = 0 \text{ (non-dendritic case)} \end{cases} \tag{D1}$$

This relationship has always been required to compute the absorption of solar radiation from the equations of Section 2.4.9. In the dendritic case, the transformation (S_i, δ_i) => (S_i, d_i) is bijective. The inversion of this relationship lead to:

$$\delta_i = \frac{10^4 \times d_i - 4 + S_i}{S_i - 3} \text{ if } d_i < 10^{-4}(4 - S_i) \tag{D2}$$

and the new evolution law of d_i was obtained by **?** using

$$\frac{dd_i}{dt} = \frac{\partial d_i}{\partial \delta_i}\frac{d\delta_i}{dt} + \frac{\partial d_i}{\partial S_i}\frac{dS_i}{dt} \tag{D3}$$

Thus, the combination of Eq. D3 with the equations from Table 1 of **?** provides the second line of Eq. 46 in this paper (which is close to the right column of Table 2 in **?** after typo corrections).

In the non-dendritic case, the transformation (S_i, g_{s_i}) => (S_i, d_i) is not bijective. When $S_i = 0$, Eq. D1 gives $d_i = 4 \times 10^{-4} \forall g_{s_i} \in [3 \times 10^{-4}, 8 \times 10^{-4}]$. As a result, the metamorphism functions from Table 1 of **?** ~~can not~~ cannot be reproduced in the space (S_i, d_i) when $S_i = 0$ (e.g. depth hoar). Then, Eq. D1 is discontinuous at the limit between dendritic ~~case~~ and non-dendritic ~~case~~cases. Indeed, the limit of the dendritic case gives $\lim_{\delta_i \to 0} d_i = 10^{-4}(4 - S_i)$. When combined with the non dendritic case, this would result in $g_{s_i} = 3 \times 10^{-4}$ while the initial value of g_{s_i} was actually higher for non-spheric particles in **?**: $g_{s_i} = 10^{-4}(4 - s_i)$. In the other cases, the inversion of Eq. D1 can provide a relationship for g_{s_i} as a function of d_i and S_i:

$$g_{s_i} = \begin{cases} 2\frac{d_i}{S_i+1} & \text{if } d_i \geq 4 \times 10^{-4}(S_i+1) \\ \frac{d_i - 4\times 10^{-4}(1-S_i)}{S_i} & \text{if } d_i < 4 \times 10^{-4}(S_i+1) \end{cases} \tag{D4}$$

Eq. D4 is actually more complex than Eq. 3 of **?** which was an incorrect simplification corresponding only to the initialization of grain size at the dendritic - non-dendritic transition. It is also not defined when $S_i = 0$ and $d_i < 4 \times 10^{-4}(S_i + 1)$. Last, the derivation of an evolution law for optical diameter was obtained by **?** in the non-dendritic case (left column of their Table 2) by considering only $\frac{dd_i}{dt} = \frac{\partial d_i}{\partial S_i}\frac{dS_i}{dt}$. This is actually inconsistent with the original formalism in the non-dendritic case as it ignores the term $\frac{\partial d_i}{\partial g_{s_i}}\frac{dg_{s_i}}{dt}$. Ignoring this term may be convenient to avoid the problems mentioned above but it affects the metamorphism of faceted crystals and depth hoar without any scientific justification whereas they are the most critical snow types for further analyses in terms of mechanical stability. Furthermore, a number of parameterizations of other processes and diagnoses in the code were also affected by the incorrect simplification of Eq. D4. The original difficulty of this translation of formalisms comes from the fact that Eq. D1 is neither bijective nor continuous. However, we considered that it is better to adapt this unpublished formula and preserve as much as possible the metamorphism laws, the parameterizations of other processes and the diagnoses relying on microstructure properties. A new metamorphism option (B21) has therefore been defined, replacing the expression from **?** by:

$$d_i = g_{s_i} \times S_i + (1 - S_i)\frac{4 \times 10^{-4} + g_{s_i}}{2} \text{ if } \delta_i = 0 \text{ (non-dendritic case)} \tag{D5}$$

This allows to replace Eq. D4 by:

$$g_{s_i} = 2\frac{d_i - 2 \times 10^{-4}(1 - S_i)}{1 + S_i} \tag{D6}$$

This way, the evolution of optical diameter in the non dendritic case can be obtained by:

$$\frac{dd_i}{dt} = \frac{\partial d_i}{\partial g_{s_i}}\frac{dg_{s_i}}{dt} + \frac{\partial d_i}{\partial S_i}\frac{dS_i}{dt} \tag{D7}$$

$$= \frac{1 + S_i}{2}\frac{dg_{s_i}}{dt} + (\frac{g_{s_i}}{2} - 2 \times 10^{-4})\frac{dS_i}{dt} \tag{D8}$$

$$= \frac{1 + S_i}{2}\frac{dg_{s_i}}{dt} + \frac{d_i - 4 \times 10^{-4}}{1 + S_i}\frac{dS_i}{dt} \tag{D9}$$

The main advantage of this new formalism is that it respects the original evolution law of $\frac{dg_{s_i}}{dt}$ as published by **?**. As a result, all parameterizations and diagnostics calibrated in the original formalism can still considered to be valid in this new formalism. Finally, the first line of Eq. 46 corresponds to the combination of Eq. D9 and Table 1 of **?**. Complementary analyses by **?** have shown that the obtained microstructure properties of B21 option are closer to the original formalism of **?** than the implementation of **?** (also referred as C13 in **?**. This is especially true when this parameterization is combined with the snow drift options. More details are available in **?**.

Similarly for wet metamorphism, the combination of Eq. D9 and Table 2 of **?** provides Eq. 51 of this paper. The transformation of wet metamorphism laws in this new formalism were neither provided in **?** nor correctly implemented.

Appendix E: Evolution of optical diameter during snow drift

The evolution of microstructure properties was parameterized in terms of dendricity δ_i, sphericity S_i and grain size g_{s_i} from simple evolution laws provided in Table 3 of **?** in which sign errors must be accounted for in the evolution of δ_i and g_{s_i} :

$$\frac{d\delta_i}{dt} = \frac{-\delta_i}{2\tau_{\text{DRIFT}}} \text{if } \delta_i > 0 \text{ (dendritic case)} \tag{E1}$$

$$\frac{dg_{s_i}}{dt} = \frac{-5 \times 10^{-4}}{\tau_{\text{DRIFT}}} \text{if } \delta_i = 0 \text{ (non-dendritic case)} \tag{E2}$$

$$\frac{dS_i}{dt} = \frac{1 - S_i}{\tau_{\text{DRIFT}}} \tag{E3}$$

In the dendritic case, the evolution of the optical diameter d_i is obtained by the introduction of Eq. E1 and E3 in Eq. D3 and computing $\frac{\partial d_i}{\partial \delta_i}$ and $\frac{\partial d_i}{\partial S_i}$ from the first line of Eq. D1. In the non-dendritic case, the evolution of the optical diameter d_i is obtained by the introduction of Eq. E2 and E3 in Eq. D7 and computing $\frac{\partial d_i}{\partial g_{s_i}}$ and $\frac{\partial d_i}{\partial S_i}$ from Eq. D5. Finally:

$$\begin{cases} \dfrac{dd_i}{dt} = 10^{-4}(S_i - 3) \times \left(\dfrac{-\delta_i}{2\tau_{\text{DRIFT}}}\right) + 10^{-4}(\delta_i - 1) \times \dfrac{1 - S_i}{\tau_{\text{DRIFT}}} & \text{if } d_i < 10^{-4}(4 - S_i) \\ \dfrac{dd_i}{dt} = \dfrac{S_i + 1}{2} \times \dfrac{-5 \times 10^{-4}}{\tau_{\text{DRIFT}}} + \dfrac{g_{s_i} - 4 \times 10^{-4}}{2} \times \dfrac{1 - S_i}{\tau_{\text{DRIFT}}} & \text{if } d_i >= 10^{-4}(4 - S_i) \end{cases} \tag{E4}$$

The first line can be slightly simplified and Eq. D6 introduced in the second line to finally obtain the equivalent discrete formulation of Eq. 67.

Appendix F: Thermodynamical functions

The air volumetric mass ρ_a is obtained by:

$$\rho_a = \frac{P_s}{R_a T_a \left(1 + \left(\frac{R_v}{R_a} - 1\right) q_a\right) + g \times z_a} \tag{F1}$$

The Exner functions at surface and at the forcing level are defined by:

$$\Pi_a = \left(\frac{P_a}{P_0}\right)^{\frac{R_a}{c_P}} \tag{F2}$$

$$\Pi_s = \left(\frac{P_s}{P_0}\right)^{\frac{R_a}{c_P}} \tag{F3}$$

where $P_0 = 10^5$Pa and the atmospheric pressure at forcing level P_a is obtained from hydrostatism:

$$P_a = P_s - \rho_a g z_a \tag{F4}$$

The saturation specific humidity at temperature T is obtained by:

$$q_{sat}(T, P_s) = \frac{\frac{R_a}{R_v} \times \frac{e_{sat}(T)}{P_s}}{1 + \left(\frac{R_a}{R_v} - 1\right) \frac{e_{sat}(T)}{P_s}} \tag{F5}$$

where the water vapor partial pressure at saturation $e_{sat}(T)$ is obtained from ~~the Clapeyron formula:~~ Eq. F6 which is an approximate integral of the Clausius-Clapeyron formula:

$$\begin{cases} \text{If } T \geq T_0 & e_{sat}(T) = \exp\left(\alpha_w - \frac{\beta_w}{T} - \gamma_w \ln(T)\right) \\ \text{If } T < T_0 & e_{sat}(T) = \exp\left(\alpha_I - \frac{\beta_I}{T} - \gamma_I \ln(T)\right) \end{cases} \tag{F6}$$

where $$\gamma_w = \frac{c_W - c_{P_v}}{R_v} \tag{F7}$$

$$\beta_w = \frac{L_v}{R_v} + \gamma_w T_0 \tag{F8}$$

$$\alpha_w = \ln(e_{sat}(T_0)) + \frac{\beta_w}{T_0} + \gamma_w \ln(T_0) \tag{F9}$$

$$\gamma_I = \frac{c_I - c_{P_v}}{R_v} \tag{F10}$$

$$\beta_I = \frac{L_s}{R_v} + \gamma_I T_0 \tag{F11}$$

$$\alpha_I = \ln(e_{sat}(T_0)) + \frac{\beta_I}{T_0} + \gamma_I \ln(T_0) \tag{F12}$$

$$e_{sat}(T_0) = 611.14\,\text{Pa} \tag{F13}$$

The ~~derivation $\frac{\partial q_{sat}(T)}{\partial T}$~~ derivative $\frac{\partial q_{sat}(T,P_s)}{\partial T}$ is obtained by:

$$\cancel{\frac{\partial q_{sat}(T)}{\partial T}} \frac{\partial q_{sat}(T,P_s)}{\partial T} = \frac{\partial e_{sat}(T)}{\partial T} q_{sat}(T, P_s) \times \cancel{\frac{1}{1 + \frac{\frac{R_a}{R_v} - 1}{1 + \frac{R_a}{R_v}\left(\frac{1}{q_{sat}(T)} - 1\right)}}} \frac{1}{1 + \frac{\frac{R_a}{R_v} - 1}{1 + \frac{R_a}{R_v}\left(\frac{1}{q_{sat}(T,P_s)} - 1\right)}} \tag{F14}$$

where $$\frac{\partial e_{sat}(T)}{\partial T} = \frac{\beta_w}{T^2} - \frac{\gamma_w}{T} \tag{F15}$$

The Richardson number is computed by:

$$\text{Ri} = g \cos \cancel{\gamma}\Theta z_u^2 \frac{\theta_{v_a} - \theta_{v_s}}{0.5\,(\theta_{v_a} + \theta_{v_s}) \max(U, U_{th})^2 z_a} \tag{F16}$$

where $$\theta_{v_a} = \frac{T_a}{\Pi_a}\left(1 + \left(\frac{R_v}{R_a} - 1\right) q_a\right) \tag{F17}$$

$$\theta_{v_s} = \frac{T_1}{\Pi_s}\left(1 + \left(\frac{R_v}{R_a} - 1\right) q_{sat}(T_1, P_s)\right) \tag{F18}$$

$$\tag{F19}$$

The wet bulb temperature T_w^* (°C) is computed by Eq. F20:

$$T_w^* = \frac{\gamma \cdot T_a^* + \frac{\partial e_{sat}(T)}{\partial T} \cdot T_d^*}{\gamma + \frac{\partial e_{sat}(T)}{\partial T}} \tag{F20}$$

where the slope of the saturation vapor pressure curve $\frac{\partial e_{sat}(T)}{\partial T}$ is given by Eq. F15 ; the dew point temperature T_d^* (°C) is parameterized by

$$T_d^* = [116.9 + 237.3ln(e)]/[16.78 - ln(e)] \tag{F21}$$

and the psychrometric constant γ (in kPa K^{-1}) is obtained by:

$$\gamma = \frac{c_{P_m} \cdot P_s}{0.622 \cdot L_v} \tag{F22}$$

where c_{P_m} is the specific heat capacity of moist air at constant pressure (in kJ kg^{-1} °C^{-1}).

Appendix G: Functions in the parameterization of shear resistance

$$\begin{cases} \text{If } S_i > 0.8 \text{ and } h_i \in [3,5]: & \mathcal{C}_1(d_i,S_i) = 1.05 \\ \text{else:} & \mathcal{C}_1(d_i,S_i) = \begin{cases} 0.45 + 0.7S_i & \text{if } S_i < 0.25 \\ 0.625 + 1.0 \cdot (S_i - 0.25) & \text{if } 0.25 \leq S_i < 0.5 \\ 0.875 + 0.6 \cdot (S_i - 0.5) & \text{if } 0.5 \leq S_i < 0.75 \\ 1.025 + 0.5 \cdot (S_i - 0.75) & \text{if } 0.75 \leq S_i \end{cases} \end{cases} \tag{G1}$$

$$\mathcal{C}_2(d_i,S_i) = \begin{cases} 1 - 0.4\delta_i & \text{if } \delta_i < 0.25 \\ 0.9 - 0.4 \cdot (\delta_i - 0.25) & \text{if } 0.25 \leq \delta_i < 0.5 \\ 0.8 - 0.8 \cdot (\delta_i - 0.5) & \text{if } 0.5 \leq \delta_i < 0.75 \\ 0.6 - 0.6 \cdot (\delta_i - 0.75) & \text{if } 0.75 \leq \delta_i \end{cases} \quad \text{where } \delta_i = \frac{d_i \times 10^4 - 4 + S_i}{S_i - 3} \tag{G2}$$

$$\begin{cases} \text{If } d_i \leq (4 - S_i) \times 10^{-4}: & \mathcal{C}_3(d_i,S_i) = 1 \\ \text{else:} \begin{cases} \text{If } g_{s_i}(d_i,S_i)(d_i,S_i) \leq 4 \times 10^{-4} - 10^{-4}S_i: & \mathcal{C}_3(d_i,S_i) = 1 \\ \text{else:} & \mathcal{C}_3(d_i,S_i) = 1 - 530 \cdot (0.8 - 0.2s) \cdot (-4 \times 10^{-4} + g_{s_i}(d_i,S_i) + 10^{-4}S_i) \end{cases} \end{cases} \tag{G3}$$

where $g_{s_i}(d_i,S_i)$ is defined by Eq. D6.

$$\begin{cases} \text{If } \frac{w_i}{w_{i\ \max}} < 0.9: & \mathcal{C}_4(w_i,\rho_i) = \begin{cases} 1 + \frac{w_i}{w_{i\ \max}} & \text{if } \frac{w_i}{w_{i\ \max}} < 0.1 \\ 1.1 - 2.35(\frac{w_i}{w_{i\ \max}} - 0.1) & \text{if } 0.1 \leq \frac{w_i}{w_{i\ \max}} < 0.3 \\ 0.63 - 0.4(\frac{w_i}{w_{i\ \max}} - 0.3) & \text{if } 0.3 \leq \frac{w_i}{w_{i\ \max}} < 0.9 \end{cases} \\ \text{else:} & \mathcal{C}_4(w_i,\rho_i) = \max\left(0.15, \min\left[0.35, (\rho_i - w_i) \times 10^{-4}\right]\right) \end{cases} \tag{G4}$$